ACTUALIDAD EN COMPUTACION,
AUTOMATIZACION DE LA OFICINA,
PROCESAMIENTO DE LA PALABRA,
Y TELECOMUNICACION DIGITAL

MI MUNDO INFORMATICO

VOLUMEN VI - Nº 169 - 1a. Quincena de Agosto de 1988

Precio: A 10

INFORMATICA Y EDUCACION

Hemos tenido una quincena activa en el tema educativo: por un lado las Primeras Jornadas de Soft Educativo Argentino y por otro el Seminario "Herramientas Informáticas para la Educación" que se efectuó en Buenos Aires y Mendoza en la que se tuvo oportunidad de escuchar a algunas figuras importantes de EE.UU. en el tema. Su participación dio, en general, la impresión de la complejidad del tema y los moderados logros que se han obtenido en un país que dispone mucho más recursos que el nuestro.

George Tressel, responsable del desarrollo de materiales de educación para la escuela primaria y media de la National Science Foundation, expresó que la primer herramienta de tecnología educativa moderna fué el fonógrafo que hizo decir a su inventor, Thomas Alva Edison, que su gran aplicación sería en la educación, porque se podrían grabar a los grandes docentes del mundo. En la década del '40 aparecieron las películas de 16 mm., en el '50 la televisión, en el '60 los satélites y en el '70 la computadora. En todos estos casos se prometía una revolución. Algunos pensaban que se estaba ante grandes cambios y que el problema residía en la adecuada capacitación del docente para su uso, pero en la práctica esto no se dio. Consideró a la computadora una herramienta diferente en posibilidades a las anteriores, pero que no reemplazaría al maestro.

El Profesor Alfredo Bork de la Universidad de California enumeró los errores que se han cometido en la introducción de la computadora en la escuela. Entre ellos citó la falta de planificación, demasiada filosofía y pocas experiencias empíricas, demasiado énfasis en el hardware en detrimiento de los aspectos pedagógicos, lenguajes inadecuados como Basic y Logo (sus desarrollos los efectúa en Pascal), inadecuada capacitación, la producción de material fraccionado con poca coherencia y la de no tener una visión clara de cuál debe ser el futuro de la educación. Sobre las características del software educativo dijo que tiene que apuntar a un aprendizaje individualizado, porque los alumnos son diferentes y esto no se reconoce en la enseñanza; el otro factor es la interactividad en el aprendizaje, que tiene que ser significativa en el sentido de que el que aprende pueda apreciar el avance en la comprensión del tema.

Estas opiniones prueban que la computadora es una herramienta educativa cuya potencialidad no se obtiene en forma simple y automática.

Quizás, con la computadora se tenga la fantasía de reproducir a nivel de cada estudiante la forma dialogada con que en su famosa Academia Platón transmitía su sabiduría a sus discípulos. Independiente de que este objetivo pueda alcanzarse algún día hay realidades en nuestra sociedad en las que la educación tiene que jugar un importante papel. Y es que en el mundo que se avecina la computadora se irá convirtiendo en una herramienta generalizada, donde cada vez más gente deberá utilizar la computadora en su trabajo. Por eso, uno de los objetivos que debería ser claro es que la educación debe contribuir a que nuestros futuros adultos se puedan integrar a ls nuevas realidades que conforman una sociedad moderna.

REPORTAJE AL LIC. ALEJANDRO OLIVEROS

SADIO: "Contribuir al desarrollo de la Informática y sus profesionales"



SADIO fue fundada en 1960, orientada a la Investigación Operativa, área que se extiende en 1974 al tema informático. SADIO se ha caracterizado por una creciente actividad reflejada a través de las Jornadas JAIIO, sus cursos, grupos de interés, su participación en el Centro de Documentación sobre Informática, etc.

Recientemente hubo una renovación parcial de su Consejo Directivo, el Dr. Hugo Moruzzi, presidente saliente, después de cuatro años de una tarea altamente positiva en el crecimiento de SADIO, ha sido reemplazado por el designado Lic. Alejandro Oliveros, con quien MI dialogó.

¿Qué significa SADIO en la comunidad informática?

SADIO es la asociación que involucra al conjunto de los profesionales de la informática con el objetivo de contribuir al desarrollo de la informática y sus profesionales. Al mismo tiempo, mantiene valiosas relaciones con sectores como las Cámaras comerciales, Asociaciones de dirigentes o egresados de Universidades, etc. Al principio de la actividad el único lugar común a los profesionales de la informática fué SADIO, pero con el tiempo han aparecido asociaciones preocupadas por aspectos específicos con las cuales nosotros nos complementamos. SADIO se enorgullece de contar entre sus socios a miembros directivos de casi todas las sociedades que están involucradas en el tema informático. Esta estructura no es específica de la informática, también se presenta en otras comunidades profesionales

¿Cúal será el futuro accionar de SADIO?

Lo primero es consolidar las actividades que viene desarrollando SADIO, en especial las ya clásicas JAIIO, la actividad de los grupos de interés, y los cursos. El segundo aspecto será el crecimiento numérico de sus asociados, apuntando a reflejar también numéricamente lo que significa SADIO en la comunidad informática.

El tercer punto es un tema en el cual venimos trabajando desde hace tiempo y es la de "incrementar la oferta de servicios", por darle un nombre de marketing. Queremos aumentar la cantidad de beneficios que el socio recibe de SADIO. Actualmente dispone de una ventajosa posibilidad de participación en las JAIIO, tiene acceso a cursos de calidad a precios muy accesibles, porque el objetivo de SADIO no es la rentabilidad de los mismos, recibe el "Noticiero" en el cual tiene información de SADIO, y de la actividad informática en general. Puede acceder a información a través de los servicios del CEDINFOR.

También se le ofrece la posibilidad de participar en los grupos de interés.

Un tema que consideramos prioritario es la reaparición de la revista de SADIO, ya que es cada día mas notoria la falta de una revista específica, profesional, de informática. En tal sentido, hay una resolución de la Comisión Directiva para implementar los pasos necesarios para el lanzamiento de la revista, próximamente haremos el anuncio de su fecha de aparición.

¿Qué grupos de interés tienen actualmente en funcionamiento?

Los grupos de interés que actualmente están funcionando son Informática Biomédica que tiene la representación del IMIA. ASINDOC, Asociación de Informática y Documentación que está activamente involucrado en el Congreso Internacional de Documentación que se va hacer en Mar del Plata en diciembre. Otro grupo de interés es el de CAD/CAM que está funcionando con niveles de actividad muy importante. El de Educación viene trabajando hace muchos años convocando a educadores e informáticos y luego tenemos el de Sistemas Expertos que es un grupo relativamente nuevo pero al que asisten una cantidad importante de interesados.

Estamos impulsando la formación de nuevos grupos. Hay socios interesados en crear un grupo que trate en profundidad la problemática de la utilización de la computación como herramienta estratégica en los negocios, estadística, simulación, etc. Los nuevos grupos de interés irán surgiendo de las inquietudes de los asociados, en principio siempre hemos escuchado, con ánimo positivo, sugerencias de cosas que podríamos hacer.

Nuestra esperanza sería en el futuro tener jornadas anuales de carácter nacional de los grupos de interés y que su actividad se vea difundida a través de boletines regulares.

Uno de los aspectos que se le cuestiona a SADIO es que su actividad es capitalina

Es cierto que SADIO tiene unaactividadfuertementecapitalina, pero estamos empeñados en proyectarnos al interior y hemos desarrollado algunas actividades que muestran que esto es viable. Las XV JAIIO en 1985 las hicimos en Bahía Blanca en la Universidad del Sur, con buena repercusión. Hemos hecho importantes cursos en distintos puntos del interior. Recientemente dimos cursos en Viedma y Rosario, en este último tuvimos más de 150 asistentes. Todo esto, que si bien son hechos aislados, muestra la factibilidad de proyectarnos al interior en la medida que tengamos apoyos locales. Actualmente estamos diseñando una propuesta para hacer una contribución importante en el tema de capacitación al personal de informática de las provincias porque vemos que tienen menos posibilidad, que el sector privado de asistir a cursos en la Capital.

SADIO tiene presente el objetivo de abarcar el país, por eso pensamos en el futuro hacer nuevamente la JAIIO en el interior.

Y con respecto a conexiones internacionales

Con respecto a nuestros contactos internacionales estamos trabajando en la creación de una red de representantes en el exterior que son útiles para poder traer personalidades extranjeras. Por otra parte participamos en el comité del IFIP de países en vías de desarrollo. Si bien integramos comisiones técnicas que nos permiten mantenernos actualizados, nos resulta muy problemático conseguir fondos que nos permitan enviar a gente para capacitarse. Hay más apoyo para enviar a alguien a exponer un tema en el exterior que mandarlo para adquirir conocimientos.

Ustedes están trabajando en concretar la próxima JAIIO que se desarrollará conjuntamente con la XIV Conferencia Latinoamericana de Informática

Así es, del 26 al 30 de setiembre haremos nuestra versión XVII y esto habla de una continuidad de años que no en muchos casos se puede exhibir en la Argentina.

Sobre la temática y las personalidades con que vamos a contar, creo que es mejor que se dirijan al Dr. Jorge Vidart, responsable del comité de programa, él les podra dar los detalles. Yo solamente voy a decir que la reacción que observamos ante nuestra próxima JAIIO ha sido sorprendente. Se recibieron 150 trabajos con una cantidad apreciable de trabajos extranjeros, esto da un idea de la repercusión de la JAIIO en el área latinoamericana.

El perfil del participante en las JAIIO es el del profesional que le preocupa su perfeccionamiento y crecimiento en informática. La XVII JAIIO ofrecerán un paquete importante de alternativas a través de cuatro carriles principales: Las sesiones de trabajos seleccionados; cursos de actualización para afrontar al desarrollo profesional; reuniones de intercambio de opinión de grupos de interés, y las conferencias plenarias. O sea son distintos andariveles que ofrecerá la JAIIO para que aquellos interesados por su desarrollo profesional encuentren puntos de interés. Estas diferentes actividades estarán cruzadas con los intereses temáticos de los asistentes que podrán escuchar desde

las aplicaciones de inteligencia artificial en manufactura hasta el de lógica difusa.

VIRUS EN LAS COMPUTADORAS

A mediados de la década del '70, la red de computadoras de un centro de investigaciones de Silicon Valley fue asaltada por un programa que se cargó a sí mismo en un puesto de trabajo ocioso, inhabilitó el teclado, dibujó figuras al azar en la pantalla y monitoreó la red buscando otros puestos ociosos que invadir.

A principios de setiembre de 1986, un intruso con talento "infectó" un gran número de sistemas de computación en el área de San Francisco que abarcaba nueve ciudades, quince compañías de Silicon Valley, nueve puestos de Arpanet y tres laboratorios administrados por el gobierno. El intruso dejó tras de sí un programa de admisión recompilado. Su meta era, aparentemente, dañar el mayor número de computadoras invadidas que pudiera; no obstante no se causaron daños.

En diciembre de 1987, un mensaje de Navidad que se originó en Alemania Occidental, se propagó por la red Bitnet a las máquinas de IBM de los EE.UU.. El mensaje contenía un programa que visualizaba la imagen de un árbol de Navidad y enviaba copias de sí mismo a todo el que figurara en la lista de distribución de correos del usuario en cuya máquina corría. Este prolífico programa atoró rápidamente la red con un número de copias que crecían en progresión geométrica. Finalmente hubo que clausurar la red hasta que todas las copias fueron ubicadas y expurgadas.

Durante dos meses, en el otoño de 1987, un programa sigilosamente incorporó copias de sí mismo en los programas de las computadoras personales de la Universidad Hebrea de Jerusalén. Fue descubierto y desmantelado por el estudiante, Yuval Rakavy, quien advirtió que ciertos programas se hacían más largos sin razones aparentes. Aisló el código errante y descubrió que en cierto viernes trece, una computadora en funcionamiento disminuía su actividad en un 80% y que el viernes 13 de mayo de 1988 iba a destruir todos los archivos. Esa fecha indicaba el cuadragésimo aniversario del último día en que Palestina funcionó como estado político independiente. Ravaky diseñó otro programa que detectaba y suprimía todas las copias del programa errante que encontraba. Aun así, no podía tener la certeza total de que lo suprimiría.

Estos cuatro incidentes ilustran los tipos de programas más importantes de los que atacan otros programas en la memoria de una computadora. El primer tipo es un "gusano", un programa que invade un puesto de trabajo y lo incapacita. El segundo es la llamada "bomba lógica", un programa que aparentemente cumple una función útil como el de acceso autorizado, pero que contienen un código oculto que realiza una función no deseada y dañosa. Se lo llama también "caballo de Troya", nombre inspirado en el legendario caballo de madera ideado por los griegos para invadir la ciudad de Troya. El tercer tipo es una "bacteria", un programa que hace réplicas de sí mismo y se introduce en el "host", mediante el "vaciado" de la capacidad de memoria y de procesamiento. El cuarto es un virus, un programa que incorpora copias de sí mismo en el código máquina de otros programas y que, cuando se llama esos programas, produce estragos a la manera de la "bomba lógica".

Se pueden citar numerosos incidentes en los que la información alojada en las computadoras fue atacada por estos programas destructores. Un centro médico en EE.UU. perdió el 40% de sus registros cuando un programa maligno se introdujo en sus sistemas. Los estudiantes de la Universidad de Lehig perdieron una gran cantidad de trabajos cuando un virus borró los diskettes insertados en las PC de la institución. Algunos programas disponibles públicamente en tableros electrónicos de noticias, destruyeron la información almacenada en los discos de las computadoras que los leían. Un artículo reciente del New York Times habla de la creciente preocupación de los administradores de redes, vendedores de software y usuarios de PC por esta forma de vandalismo electrónico.

Todos estos ataques "virales" recientes han atraído la atención sobre el problema más general de la seguridad, un tema de gran complejidad que fascina a los investigadores desde hace largo tiempo. Esta posibilidad de programas perniciosos que se propagan a través de sistemas de archivos es conocida desde hace veinticinco años, por lo menos.

Es importante tener presente que estos "gusanos", bombas lógicas", "bacterias" y "virus", son programas diseñados por seres humanos. Si bien el examen de estas amenazas trae a colación intrigantes problemas técnicos, no debemos olvidar que la raíz del conflicto reside en los programadores que efectúan esos desmanes protegidos por el anónimo que proporcionan muchos sistemas de computación.

Centrémonos ahora en los virus. Un virus es el segmento de un código que se ha incorporado al cuerpo de otro programa "infectándolo", cuando se ejecuta el código del virus, éste se ubica en otros programas infectándolos. con el paso del tiempo, el número de programas infectados puede ser considerable. Los virus pueden diseminarse con notable rapidez. Un estudio realizado en la Universidad de Cincinnati demostró que un simple programa virus puede propagarse a casi todos los rincones de una computadora que funciona normalmente, en cuestión de horas. La mayor parte de los virus contienen un indicador que les permite reconocer las copias de sí mismo; con ello evitan que los descubran, ya que de otro modo, algunos programas se harían notablemente más largos tras sucesivas infecciones. Los actos destructivos vienen más tarde; cualquier copia del virus que corra después de la fecha señalada, provocará la acción no deseada.

Un programa "caballo de Tro-



EDITORIAL EXPERIENCIA

Avda. Pte. Roque Saenz Peña 852, 5º Piso Of. 514 - 1035 - Cap. Tel. 49-1891



DIRECTOR- EDITOR
Simón Pristupin

CONSEJO ASESOR
Lic. Jorge Zaccagnini
R. ul Montoya
Cdor. Oscar S. Avendaño
Dr. Antonio Millé
Ing. Alfredo R Muñiz Morer o
Cdor. Miguel Martin
Juan C. Campos
Ing. Enrique Draier
Ing. Jaime Godelman
C.C. Paulina Frenkel

REDACCION
Luis Pristupin

COMPOSICION
Vientocur

DIAGRAMACION
Linea y Papel

Precio del ejemplar: A 10
Precio de Suscripción: A 200

Suscripción Internacional:
América
Superficie: U$S 30
Vía Aérea: U$S 60
Resto del mundo
Superficie: U$S 30
Vía Aérea: U$S 80

Registro de la Propiedad Intelectual No. 37.283

Durante décadas la barrera de los 4.90 m permaneció inalcanzable. En 1962 este record fue finalmente superado y ahora sobrepasa los 5.80 m.

Esta marca se alcanzó no solo por el esfuerzo de los atletas sino también por el avance de la tecnología. La vara de fibra de vidrio había sido introducida al salto de garrocha. Este hecho mejoró la performance.

ADR PERFORMANCE SOFTWARE
TECNOLOGIA QUE LLEVA A LA PRODUCTIVIDAD A NUEVAS ALTURAS

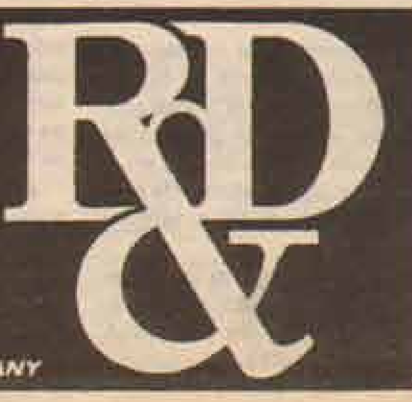
Casi 30 años trabajando para optimizar el uso de recursos en las empresas, dieron como resultado productos como: Roscoe, incrementa la productividad de los programadores que trabajan on-line; The Librarian, reduce el tiempo que demanda el mantenimiento de programas; Ideal, aumenta en forma contundente la productividad mediante la modernización del proceso de programación; Datacom/DB, ofrece la indispensable flexibilidad relacional a un ambiente de producción. Y, actualmente nuevas tecnologías como Case y el sistema experto Mindover MVS.

Todo esto es con un solo objetivo: incrementar la productividad, por eso si su objetivo es el mismo, cambie su tecnología al software de performance de ADR.

LIBERE SU POTENCIAL

Estas soluciones están disponibles para los equipos IBM 43xx, 30xx, 93xx y compatibles, bajo los sistemas operativos DOS/VS al VSE/SP y OS/VS1 al MVS/A.

TECNOLOGIA Y SERVICIOS EN SOFTWARE DE AVANZADA



R&D S.A., Representante Exclusivo de **APPLIED DATA RESEARCH** **ADR**
Lavalle 1616, 3er. Piso, (1048) Buenos Aires, Argentina, Tel. 46-6881/2

AN AMERITECH COMPANY

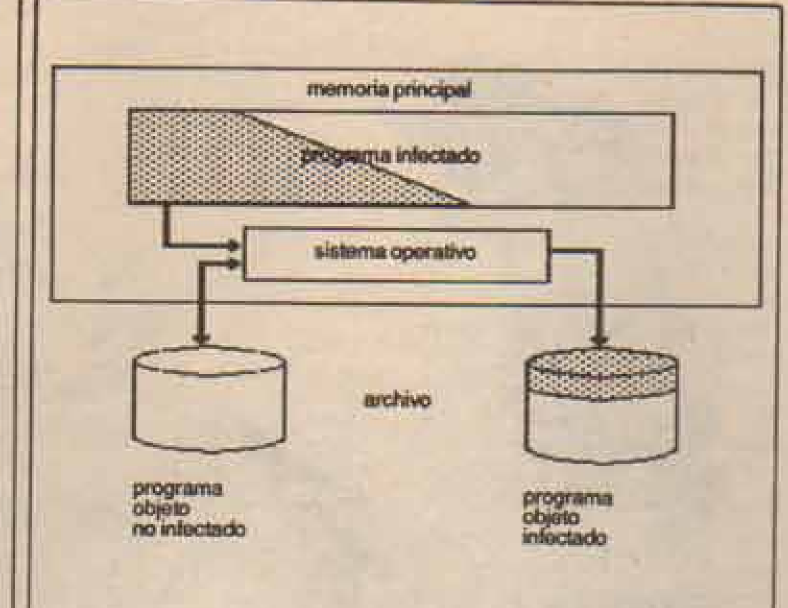
Miembro de la Cámara de Empresas del Software (C.E.S.)

ya" es el medio más común para introducir un virus. Es posible introducir un "caballo de Troya" invisible en un compilador, el cual implanta otro caballo de Troya en un programa determinado, durante la compilación.

Un virus que adopte la forma de sentencias insertadas en el lenguaje de alto nivel de un programa — esto es, en el archivo fuente— puede ser detectado probablemente por un experto que lea el programa, pero es muy difícil descubrirlo en un programa de un sistema grande.

Para combatir estos virus, verdaderas "bombas lógicas", se han introducido ciertos rasgos especiales en el hardware como, por ejemplo, una memoria virtual que restringe un programa a una región limitada de la memoria. Todos estos rasgos tienen como base el principio del menor privilegio, que reduce el conjunto de archivos objetos accesibles al mínimo indispensable para que el programa cumpla su función.

Un hardware con protección de memoria puede reducir los riesgos, pero el virus puede todavía propagar programas legítimamente accesibles, inclusive a porciones del sistema operativo. La velocidad de propagación puede ser reducida, pero no detenida por la memoria virtual. La mayor parte de las microcomputadoras personales es vulnerable, porque no tienen protección de la memoria; un programa ejecutable tiene acceso a la memoria del hardware o a la memoria en disco. Una red de PC es más vulnerable todavía, porque cualquier computadora puede propagar una copia infectada de un programa a cualquier otra máquina de la red.



COMO OPERA UN VIRUS

Un programa infectado por un virus (parte grisada) que se carga y se ejecuta en la memoria principal de una computadora, puede infectar otro programa ejecutable (objeto) en el archivo de memoria en discos, solicitando secretamente al Sistema Operativo de la computadora, que anexe una copia del código del virus al programa objeto, generalmente al comienzo de éste. La infección alarga levemente al programa objeto.

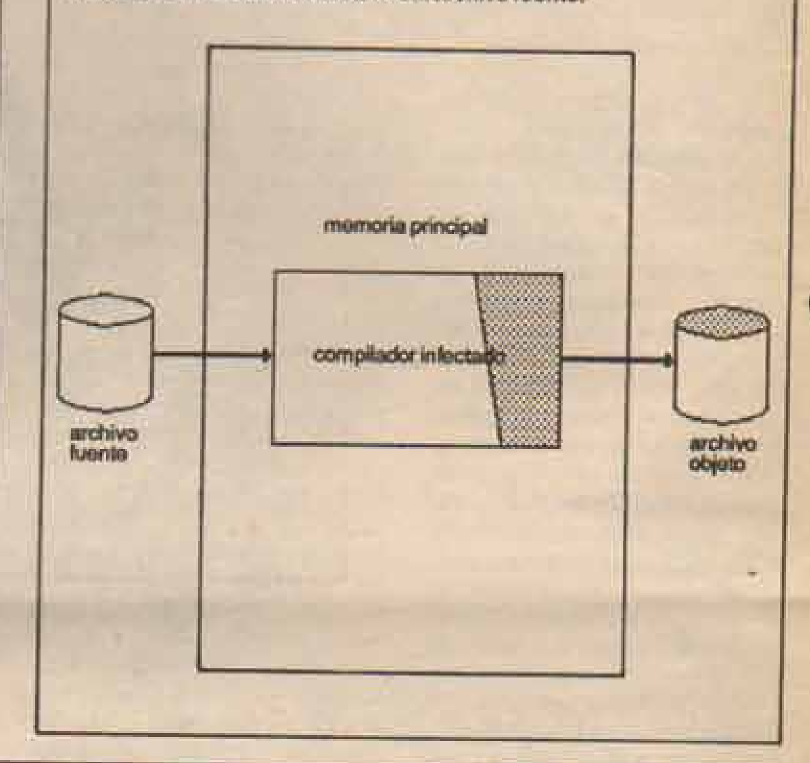
Cuando el programa así infectado se carga en la memoria y posteriormente se lo llama, el virus que contiene asume el control y efectúa acciones ocultas, tales como la de infectar otros programas objeto. El virus puede realizar asimismo otras funciones destructivas antes de transferir el control en el punto de entrada del programa original. El código del virus contiene un indicador que hace que no se infecte un programa ya afectado anteriormente, evitando que con el aumento de la longitud del programa original sea posible descubrirlo.

El mismo principio opera en las microcomputadoras de donde los diskettes desempeñan el papel de los programas objeto. En este caso, el virus ataca la copia del Sistema Operativo contenido en el diskette, de modo que es automáticamente incorporado tan pronto el Sistema Operativo entra en funcionamiento. Como el sistema operativo reside en ese momento en la memoria de la computadora, puede infectar a todos los diskettes que se inserten en la PC.

LA "BOMBA LOGICA" EN UN COMPILADOR

La "bomba lógica" es un código oculto (grisado en el dibujo), de un programa útil, que efectúa una acción oculta y mal intencionada. Puede copiar archivos privados de un usuario en la memoria del autor del programa infectante, atentando así contra la privacidad de la información que debería ser inaccesible para el diseñador del virus. La "bomba lógica" también puede destruir o borrar archivos, se la conoce también con el nombre de "Caballo de Troya".

Se ha sugerido que la "bomba lógica" puede detectarse examinando el archivo fuente de un programa, buscando sentencias que efectúen operaciones fuera de las especificaciones del programa. Ken Thompson, uno de los principales diseñadores de UNIX, señaló que este enfoque es fundamentalmente incompleto. Demostró como "arreglar" un compilador, introduciendo una "bomba lógica" en el archivo objeto de un programa cualquiera, como por ejemplo el que identifique y autorice el acceso, en el que cada vez que el programa infectado se vuelve a compilar, se inserta un segmento de código cuando una especial contraseña, conocida por el delincuente, es dada. Este programa pirata no puede descubrirse mediante la lectura del archivo fuente.



Muchos se están haciendo esta pregunta. Pero no piensan en linfocitos y cadenas ARN sino en bytes e instrucciones ASSEMBLER. El virus que temen no es biológico, sino informático.

¿EXISTE UNA VACUNA CONTRA EL SIDA?

Por: Lic. Alberto A. Serritella

Un artículo aparecido en una publicación del país provocó el pánico en muchos usuarios de computadoras que llevaban hasta ese momento una vida apacible.

Y la pregunta del encabezamiento en realidad es consecuencia de una previa: ¿Mi software tiene el virus?.

Antes que nada: ¿en que consiste el así llamado "Virus del software" o "Sida del Software"?. Debe especificarse que el virus informático guarda una gran analogía con el virus biológico del Sida.

En primer lugar su proceso de incubación es muy lento y mientras dure el mismo no se percibe ningún síntoma.

En segundo lugar existen portadores sanos y otros en lo que la enfermedad se ha declarado. En estos últimos el proceso es mucho más fulminante que en el Sida biológico. Es altamente probable que todo portador termine enfermo.

En tercer lugar y al igual que su hermano biológico, el Sida informático se contrae primordialmente en medios promiscuos, contra natura y que cuentan con adictos y traficantes: el mercado del software pirata. En otros casos se contrae incluso bajo receta de especialistas a traves de una transfusión de programas por un canal de comunicación. Desde este punto de vista no hay mucha diferencia entre un Modens y una Modess: ambos pueden transmitir la enfermedad.

De todas maneras debe aclararse que el virus informático no es único. Existen en realidad clases distintas.

El contador que se encontró con el muñequito tipo Pac-man había contraído en su máquina la especie más sofisticada: "el virus inteligente". Consiste en un programa tipo "demonio". Una de sus principales características es su capacidad para reproducirse.

Recordemos que uno de los "demonios" informáticos más famosos, tal vez el precursor, fue ideado por un programador que se quiso vengar de un despido previsto por parte de la empresa que lo contrataba. Dejó un programa que comenzó a operar a partir del momento en que fue dado de baja en el archivo de empleados y que terminó invadiendo toda la computadora.

La "teleología" (no me animo a hablar de "finalidad") del virus informático es atacar el mercado del software pirata. Haya sido o no la intención de quién lo programó, en la práctica el Sida del software afecta a los usuarios de programas copiados ilegalmente.

La especie "inteligente" mencionada posee una sofisticación notable. Puede permanecer oculto largo tiempo inactivo. Un buen día comienza a desprender "metástasis" pero aun nadie se da cuenta. En su etapa final y fulminante destruye los archivos del afectado y corona su obra con la aparición del Pac-man y un mensaje de corrosivo humor. Hasta el momento conocer el Pac-man ha sido un privilegio muy caro de unos elegidos. Pero no se desanime. Prosiga utilizando todo el software que le han regalado sus amigos y muy pronto será recompensado.

Pero bromas aparte tampoco es cuestión de crear mitos. Este tipo de programas es sofisticado pero de ninguna manera patrimonio exclusivo de algún genio de las computadoras. Seguramente dentro de poco tiempo han de proliferar distintas versiones.

Existe un segundo grupo de virus, que no merece exactamente ser denominado así pues no posee capacidad de reproducirse por sí mismo. Es colocado en programas usados como "cebo". Son los piratas del software los reproductores involuntarios.

Al igual que la especie "inteligente" esta especie "muda" comienza a actuar luego de un tiempo prolongado de inactividad. Su acción contaminante es muy discreta y se propaga a otros programas. A pesar de su falta de inteligencia este virus no es menos temible que el anterior. Debe pensarse que el virus "inteligente" al menos da un mensaje avisando la destrucción de los archivos. El virus "mudo" es mucho más insidioso pues hace que todo archivo se vuelva poco confiable. No se sabe si hay archivos afectados y la medida de los posibles daños.

Existe en realidad una tercer especie de virus, relativamente inocua. Se trata de un virus "aleatorio". Nadie lo programa. Se origina en una falla muy pequeña en el software que no impide la operación. Esta falla se potencia en rutinas por lo común residentes en la memoria principal y pueden producirse problemas que habitualmente consisten en "colgar" el equipo o daños manifiestos de los archivos. Si este proceso se puede o no potenciar en niveles de mayor inteligencia es un tema que da pie a relatos de ciencia -ficción.

¿Como podemos estar a salvo del virus del software?

Son posibles algunas recomendaciones:

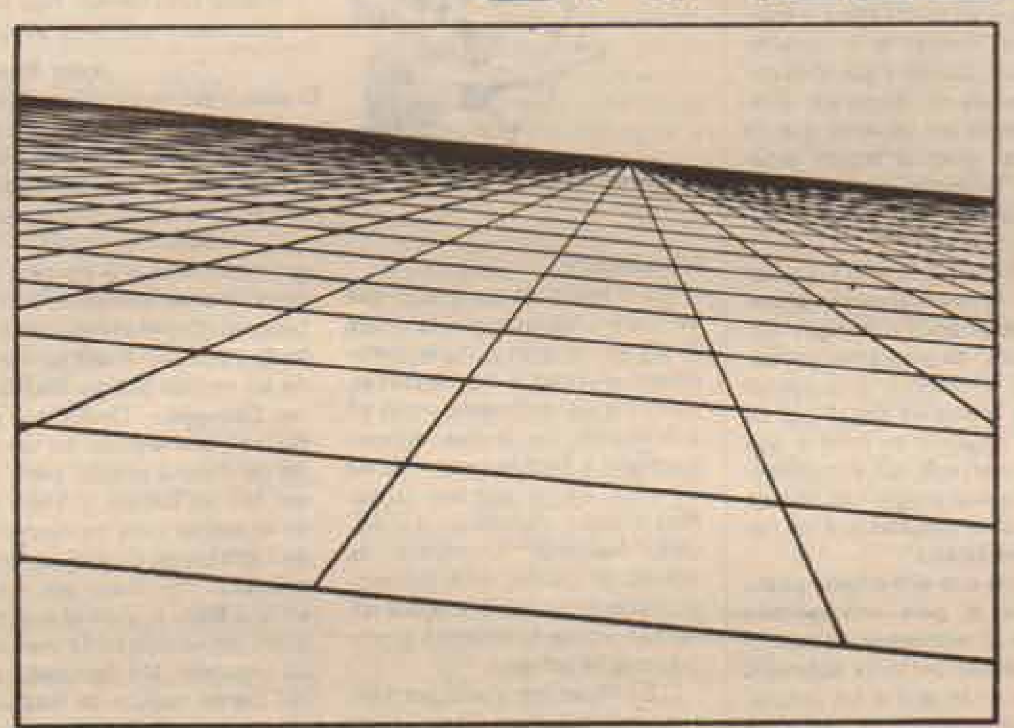
1º) Utilice siempre software original. Evite las copias ilegales

2º) Tenga en cuenta que los datos pueden ser afectados, pero no actúan como contaminadores activos. La función activa sólo puede residir en programas. Todo programa o rutina ejecutable es sospechosa. Ni siquiera están libres de sospechas los archivos ocultos del sistema operativo. Es posible que parte del virus se disimule entre datos o incluso en zonas no utilizadas de los soportes magnéticos. Pero el detonador que permite "armar" el virus siempre se encontrará en un programa.

3º) Si Ud. por curiosidad almace-

UTILIZAR ORACLE EN SU EMPRESA ES LA MEJOR DECISION INFORMATICA QUE USTED PUEDE TOMAR HOY.

ORACLE®



INVIRTIENDO EN PRODUCTOS DE ORACLE USTED **OBTENDRA:**

☐ Una drástica **reducción en los costos y tiempos de desarrollo** de sus sistemas informáticos.

☐ La posibilidad de **integrar definitivamente** sus **sistemas operacionales** con sus **sistemas de información** y sus **sistemas para la toma de decisiones.**

☐ La posibilidad de **procesar sus sistemas en cualquier computador,** utilizando el equipamiento más efectivo para cada aplicación (desde computadores personales hasta grandes computadores de cualquier marca y modelo [*]).

☐ La posibilidad de **desarrollar en computadores personales sistemas que serán procesados en grandes computadores y viceversa.**

☐ La posibilidad de **implementar sistemas distribuidos** (sistemas descentralizados que comparten información en forma dinámica cuando ello es necesario).

☐ La posibilidad de **centralizar o distribuir** el procesamiento de sus aplicaciones, en cualquier momento, **sin necesidad de modificar sus sistemas.**

☐ La posibilidad **que el procesamiento sea realizado en computadores personales utilizando dinámicamente datos residentes en su computador central** (concepto de estación de trabajo). De esta manera usted puede incorporar potencia de procesamiento en forma incremental (a costo marginal) escapando del cepo al que su computador central lo tiene sometido.

☐ La posibilidad de **utilizar más eficientemente su equipamiento actual** (el Software provisto por su proveedor de Hardware nunca será tan eficiente).

Sólo ORACLE, líder en tecnología y comercialización de Software de bases de datos en todo el mundo, le ofrece estas posibilidades **hoy.**

ORACLE le permite adoptar la estrategia de crecimiento más económica preservando al mismo tiempo sus inversiones en desarrollo de sistemas y en capacitación de personal.

La tecnología de **ORACLE** está basada en estándares de amplia aceptación y sus productos establecen nuevos estándares en el mercado informático.

Usted puede utilizar estas ventajas en beneficio de su empresa eliminando costos de oportunidad emergentes de la utilización de productos y métodos obsoletos, y de la frecuente adaptación de sus sistemas a nuevos computadores y sistemas operativos.

(*) más de 60 computadores y 14 diferentes sistemas operativos. La lista no incluye sistemas IBM S/3X.

Programa Seminarios ORACLE 1988 sin cargo

29/06 Capital Federal (*)
03/08 Córdoba
29/09 Mendoza
22/11 Rosario

(*) Durante 1988 en Capital Federal se realizarán seminarios sin cargo los días martes.

Para la inscripción o consultas dirigirse a Data S.A. - Administración de Ventas - Av. Belgrano 990 8vo. piso - Tel.:334-3132 ó 334-9081/84 int. 302

ORACLE®

COMPATIBILIDAD · PORTABILIDAD · DISTRIBUTIVIDAD

DATA S.A.

BERNARDO DE IRIGOYEN 560 - 1072 - 334-3132 334-0273 334-2282 334-7417

na software sospechosos de infección, manténgalo totalmente aislado del software sano. Tenga en cuenta que hasta un diskette "original" se puede afectar si es usado conjuntamente con el software contaminado. Es posible imaginar métodos que consiguen grabar en un diskette protegido. Incluso apague su computadora luego de usar software sospechoso.

4º) Puede realizar back-ups de los diskettes originales pero cuide de que las rutinas empleadas para efectuar la copia estén libres del virus. Utilice el sistema operativo original. Desconfíe en especial de las rutinas del tipo "Copia-Tutti".

5º) No minimice la eficacia del virus. Pueden trasmitirlo copias ilegales de sistemas operativos, bases de datos, procesadores de palabra o planillas electrónicas de apariencia inocente. Incluso los sistemas que Ud. mismo programa pueden ser contaminados por intérpretes y compiladores (Cobol, Basic, Pascal, etc.). Una sola copia ilegal puede contaminar todo su software. No ceda a la tentación por más seguro que se sienta.

6º) Aunque sus conocimientos de computación sean muy avanzados le costará mucho detectar el virus. Los sistemas de camouflage admiten una cantidad de formas.

7º) Desconfíe de los programas "anti-virus". Piense que el virus puede quitarles el control. El canario cantor que se utiliza para indicar la vitalidad de su máquina puede ser substituido por un falso canario que luego se transforme en Pac-man. Y aun sin substituir el canario, considere que el programa anti-virus sólo conseguirá detectar algunas formas del virus. Si el programador del virus ha demostrado tanta saña, piense que nada le impide insertar un "gen mutante" que haga que luego de cierto momento o en determinadas circunstancias cambie radicalmente sus modos de acción.

8º) Debido a la extensión del mercado pirata en el país, es probable que la mayor parte del software se halle contaminado. No piense que su máquina está a salvo del virus por el simple hecho de no observar nada extraño. El efecto retardado es una de las principales características del virus. Este efecto retardado depende de condiciones aleatorias imprevisibles para el usuario.

9º) No se desespere. El virus no ataca a la computación. Tan solo a las piratería. Vuelva a compilar sus programas con un compilador original o copia no contaminada del original. Tenga en cuenta que el virus se oculta en los programas "objeto", no en las "fuentes".

Revise la coherencia de sus archivos de datos con una validación adecuada.

10º) Si Ud. se halla conectado a una red informática o establece conexiones telefónicas por medio de su computadora, entonces intercambie datos, pero nunca rutinas o programas por tal tipo de medio.

Si Ud. toma en cuenta puntillosamente los consejos anteriores, puede descansar tranquilo pues está razonablemente a salvo del virus.

Todo parece indicar que el virus llegó desde el extranjero (aunque se sospecha que existen algunas especies vernáculas del tipo "mudo"). Como método para combatir el mercado pirata dará pie a polémicas. Pero es probable que termine convirtiéndose en una ayuda importante para la producción nacional de software. Quienes nos dedicamos al desarrollo de sistemas informáticos, hemos apelado siempre a métodos de protección "angelicales": la copia pirata no funciona y punto. Siempre consideramos que es poco serio agregarle alguna "maldad" a la copia no autorizada, debido a que el mecanismo puede ser disparado accidentalmente por usuarios que no pretendan efectuar copias ilegales. Es evidente que hubo alguien que no pensó de esta manera.

Muchos usuarios no compran programas que realmente necesitan pues pueden conseguir copias "gratis" de programas parecidos. En la práctica la gratuidad de tales copias es engañosa: se termina pagando en hard lo que se ahorra en soft, con el agravante de emplear programas de baja calidad o no adaptados a las necesidades locales.

Se dice que el mercado pirata ocasiona al país una pérdida anual de 32 millones de dólares.

El método del virus apunta a proteger, más que a los programas de producción propia, a los programas hechos por otros y que no se hallaban protegidos contra copia. Realmente es una idea totalmente sorprendente y de una eficacia optimizada.

Paradójicamente llena un vacío legislativo en lo referente a protección de software.

Una cuestión que puede plantear algunas dudas es la siguiente: El software de base original, ¿no puede también estar afectado por el Sida? Se supone que los productores de software serán los primeros en velar por la no contaminación de los sistemas propios. Y que cuidarán que entre sus programadores no se infiltre nadie dispuesto a diseminar el virus. Disponiendo de los programas fuentes originales, de editores, compiladores, y rutinas de base, es posible producir software razonablemente a salvo de infiltraciones del virus. Mi método preferido es trabajar, aun en programas de alta tecnología, con métodos casi artesanales. De tal forma, se conserva el control y la calidad de lo realizado.

Debo agregar por último y contestando la pregunta inicial, que, en el caso del software del cual soy autor (procesador de palabras, mailing, etiquetador, manejadores de impresión, etc.) lo protejo con una vacuna totalmente efectiva. La misma no puede evitar la contaminación. Pero produce la muerte de la copia afectada evitando que se convierta en portadora.

De aquí en más agregaré a las garantías que otorgo sobre el software una que resulta tan insólita como necesaria: "Garantido contra el virus"

A Propósito del Lenguaje C, Eratóstenes, y Alicia en el País de las Maravillas

Casi todo lo que se puede hacer en otros lenguajes se puede hacer en C y más fácilmente, de forma más eficiente. En C pueden hacerse programas que con los demás no (1), o cuyo costo de desarrollo es muchísimo mayor. También demostraremos Ud. y yo por enésima vez que un ejemplo vale más que mil palabras. En el intento, Lewis Carroll nos ayudará a ubicar seis y medio millones de piedras donde cabe un millón, de la mano del gran filósofo y matemático Eratóstenes. Este artículo podría llamarse 'Encuentre en un rato medio millón de números primos en su PC'.

Javier Blanqué

El artículo presupone el conocimiento de programación estructurada y un lenguaje de alto nivel como Algol, PL/1, C o Pascal, o de lo contrario muchas ganas de aprender por parte del lector.

Introducción

La mayoría de las quejas que se hacen respecto del lenguaje C (2), se refieren a implementaciones antiguas o muy limitadas, definidas generalmente como un subconjunto de lo que hicieron Kernigan y Ritchie hace más de 15 años, de las que hoy no vamos a hablar. Actualmente el Instituto Nacional Americano de Standards (ANSI) está definiendo una norma para C, a la que adherirán varios fabricantes y productores de software.

¿Es necesario repetir por milésima vez que Unix (3) se afirma en el mundo como standard en estaciones de trabajo científicas, de ingeniería, y ambientes de supermicros multiusuarios, y todos tienen su versión, IBM : AIX, Apple : A/UX, Microsoft : Xenix, Microport-ATT-Olivetti : Unix, DEC : Ultrix?

Que el lenguaje base de Unix es C, que *los fuentes de Unix escritos en C están disponibles* desde ATT -algo de importancia estratégica para quienes deciden-, y que la cantidad de ayudas al desarrollo de programas que existen en ese Entorno Operativo es descomunal -Lenguajes de comandos (B y C SH), Lenguaje C, Compiladores de Compiladores (YACC), administradores de Proyectos y Programas (Make), generadores de reportes (AWK), buscadores por contenido (GREP), depuradores a nivel fuente (SDB), y administradores de bibliotecas (SCCS)-, por lo tanto, hoy tampoco hablaremos de Unix. Veremos algo de lo que se puede hacer con C en computadores personales.

Existen excelentes compiladores disponibles tanto para el ambiente de IBM PC/AT/Compatibles, como para IBM PS/2 (Metaware C 1.4, Turbo C de Borland, Quick C y MS-C 5.0 de Micro-Soft) y Apple Macintosh (MPW 2.0 de Apple y LightSpeed C de Think Technologies). Incluso existen excelentes intérpretes y compiladores incrementales de C (Instant C y C-Terp entre otros). (4)

Estos compiladores ya no vienen aislados, ni son difíciles de utilizar, generalmente incluyen en el paquete un buen editor de programas fuente tipo pantalla integrado y otros utilitarios, aún Debuggers (Depuradores). Respecto de éstos, Microsoft se mantiene en ventaja con su *Source Level Debugger* CodeView, pero Borland está a punto de sacar uno de factura propia, para vender con su TurboC, y Think está en lo mismo para su versión 3.0 de LightSpeed C -que compila a cincuenta mil líneas por minuto en una MAC II, y en el que cualquier linkedición tarda menos de un segundo-. Un depurador a nivel fuente reduce la resolución de problemas difíciles a niveles mucho más manejables.

Pequeña descripción del lenguaje

El lenguaje C está orientado al desarrollo de sistemas y grandes programas comerciales, científicos y técnicos, es estructurado y recursivo, tiene opciones de alto y bajo nivel, como manejo de estructuras de datos autorreferentes (como árboles y redes), pasaje de funciones como parámetros y manejo de campos de bit. Tiene un juego de operadores y tipos de dato que cubren todo el espectro de lo que necesita quien va a desarrollar un programa que requiere brevedad en el código escrito, reducida cantidad de código generado (sólo 20-30% más en que assembler), y alta velocidad de ejecución. Tiene verificación de funciones y tipos de dato (Strong Typing), que puede habilitar o no el programador a elección, y la exactitud en manipulaciones matemáticas depende de las bibliotecas que maneje, por ejemplo en la MAC, con LightSpeed C *no hay ningún problema de redondeo*, pues sigue la norma standard IEEE para aritmética de punto flotante (SANE) con posibilidad de cálculo sin overflow de números reales de hasta ±10±4932. Algo semejante sucede en Unix cuando los programas se encadenan con la librería 'math'. La asignación dinámica de memoria es manual por una cuestión de eficiencia (a nadie se le impide escribir su propio 'Garbagge Collector'), pero el programador dispone de una buena librería para administrar el espacio que necesita. Y las librerías de uso general y específicas de que dispone son amplias y ampliables, muchas de ellas *con todos sus fuentes disponibles.*

ANSI C (5)

Existen una serie de mejoras hechas sobre el C standard de K&R, y que funcionan en la mayoría de los compiladores nombrados, que adoptó ANSI, entre ellas las más importantes son la expansión de las librerías; el chequeo de funciones a través de prototipos; el chequeo de tipos de datos (que realizaba el programa Lint en Unix); la inclusión de funciones 'In Line', es decir, funciones cortas que se expanden como macros; la posibilidad de pasaje estructuras (Records en Pascal) como parámetros; la inclusión del tipo 'void', importante en los prototipos de funciones y en el pasaje y definición de variables que apunten a tipos de datos arbitrarios; y la adopción del tipo de datos 'enum' o enumerativo, por ejemplo :

typedef enum Nota { do, re, mi, fa, sol, la, si, fin };
Nota musical, escala;...
musical = re;....
for (escala = do; escala < fin ; escala++) Tocar(escala);...

Optimización (6)

El avance en el software de lenguaje C -intérpretes y compiladores-, no es sólo en extensión y opciones, sino en métodos de optimización, por ejemplo en el trabajo hecho en la parte de optimización del código y el espacio ocupado por los datos, existen varias técnicas, algunas son dependientes de la máquina y otras no.

<u>Técnicas independientes del hardware</u>

Aritmética de constantes

Evita la evaluación en tiempo de ejecución de expresiones cuyo valor puede ser extraído en tiempo de compilación. Ejemplo 'a = 0.5+2.5;' se compila como 'a = 3;'

Propagación de constantes

Convierte expresiones en asignaciones constantes, evaluando las primeras en tiempo de compilación. <u>Ejemplo</u> 'a = 1; b = a+1;' se compila como 'a = 1; b = 2;'

Eliminación de expresiones comunes invariantes

Permite remover expresiones cuyo valor permanece constante mientras se ejecutan los mismos. <u>Ejemplo</u> 'a = b + d * e; ...f = g + d * e;' se compila como -siempre que d y e no se modifiquen en el medio- 'temp = d * e; a = b+temp; f = g + temp;'

Eliminación de memoria muerta

No se asigna espacio a las variables no utilizadas, y las sentencias no alcanzadas no son tomadas en cuenta por el compilador, pueden ser tanto un error del programador, como producto de una transformación del optimizador.

Propagación de copias

Reduce la cantidad de sentencias, modificando la ubicación de las variables de las expresiones. Ejemplo 'a = b+c; d = a; e = d+f' se copila cómo 'a = b+c; e = a+f'

Traslación de código invariante en ciclos

Los bloques de sentencias que no se modifican dentro de un ciclo, se trasladan fuera del mismo. Ejemplo 'for (i = 0 ; i < 10 ; i++) tabla[i] = a + b;' se compila cómo 'temp = a + b; for (i = 0 ; i < 10 ; i++) tabla[i] = temp;'. Nota : x++ significa x=x+1.

Reducción de complejidad de operaciones en ciclos

Existen operaciones que son más intensivas computacionalmente que otras en el uso de la CPU, el objetivo es cambiarlas por otras menos intensivas. Ejemplo 'for (i = 0 ; i < 10 ; i++) tabla[i] = i * 4;' se compila como 'temp = 0; for (i = 0 ; i < 10 ; i++) { tabla[i] = temp; temp += 4; }'. Nota : x += y significa x=x+y.

Usar apuntadores en lugar de subíndices

Utilizaremos esta técnica más adelante, sólo debemos tener en cuenta que en C existe una estrecha relación entre aritmética de apuntadores y el manejo de vectores y matrices, y que existen operadores para el manejo de direcciones : 'a = &b;' significa 'a toma la dirección de b' y 'a = *b;' significa 'a toma el contenido de b'.

Existen otras técnicas de optimización

'Eliminación de variables de inducción', 'Desdoblamiento de ciclos', 'Unión de ciclos compatibles', etc., pero con lo explicado hay para entretenerse.

Técnicas de optimización dependientes del hardware

Pueden nombrarse la 'Asignación automática de variables a registros de la CPU', análogo a lo que hacemos manualmente en lenguaje ensamblador con las variables muy usadas. 'Permutación de operaciones para reducir la cantidad de registros usados en cálculos', etc.

Un Ejemplo vale más que mil palabras

Vamos a generar números primos, récordemos que un número Natural es primo si ningún número lo divide salvo el 1 y él mismo. En realidad este artículo comenzó cuando, desarrollando unas funciones de Hashing para un Núcleo de Sistema Experto -Expert System Shell- (7), necesité números primos grandes (8) y (9), entonces con el afán de no reinventar la rueda, la idea fue tomar los antiguos conocimientos de Eratóstenes sobre cómo generar números primos, técnica bastante pública y conocida, el algoritmo es más o menos como sigue :

Nosotros

Crear una Tabla grande de marcas
-Cada marca indica un Natural/Primo
Llenar elementos de Tabla con marcas

Para cada elemento de la Tabla
ver si tiene marca
SI : Sacar las marcas de todos los múltiplos del mismo.
NO : Pasar al siguiente elemento.

Al final del procedimiento, los elementos que tengan marcas son primos.

Eratóstenes

Construir una Criba o malla con fondo
-Cada piedra indica un Natural/Primo
Llenar los huecos de la Criba con piedras

Para cada hueco de la Criba
ver si tiene piedra
SI : Sacar las piedras de todos los huecos a igual distancia del mismo.
NO : Pasar al siguiente hueco.

Al final del procedimiento, los huecos que tengan piedras son primos.

Sin embargo los problemas comienzan cuando la tabla es de más de 16383 elementos enteros (short int), o de más de 32767 elementos de 8 bits (char), pues el Compilador que usamos -LSC versión 2.11- no permite Tablas de más de 32KB, ni tampoco Segmentos de Datos de más de 32KB, con lo que el límite de la forma clásica de encarar la criba estaba dado. El *ejemplo 1* está centrado en ese límite, corrimos el programa y obtuvimos 3512 primos, hasta el natural 32767 (En los fuentes hay estadísticas de velocidad y otros). Entonces decidimos modificar el programa y crear el *ejemplo 2*, que además de optimizar el manejo de tablas -con aumento de la velocidad de ejecución- cambiando subíndices por apuntadores, crea la tabla con una función de sistema operativo (NewPtr) que permite eliminar el límite de 32KB, y direccionar a toda la memoria RAM -de acceso aleatorio- disponible (1MB menos lo ocupado por el programa y el S/O). Corrimos el programa y obtuvimos como máximo 65052 primos, hasta el natural 815000. Esto no nos venció, ya que podíamos echar mano de nuestro viejo amigo Carroll, y del País de Alicia, desde donde sacamosde la galera el conejo y también la solución al problema -*ejemplo 3*-. El asunto estaba cocinado utilizando 'Bit Fields'. Podíamos dividir un byte en sus partes componentes, los bits. Entonces lo escribimos, lo corrimos, exprimimos la memoria, y después de más de una hora de laburo -del enano que el equipo tiene adentro-, obtuvimos cómo máximo 443481 primos hasta el natural 6480000, y basta por hoy. A continuación siguen el proyecto, fuente de símbolos, fuentes de los ejemplos, y partes del archivo de primos que fue generado, tal vez le sirvan a alguien que trabaje en teoría de números o criptografía. Nota : Comentarios en C van entre '/*' y '*/'.

NOTI CAESCO

EN LAS PARITARIAS LOS REPRESENTANTES EMPRESARIOS DEBEN NEGOCIAR CON FIRMEZA

La apertura de las negociaciones paritarias, tal como CAESCO lo había previsto, obliga a las organizaciones empresarias a ejercer un rol protagónico, reafirmando en términos no excluyentes el interés del sector.

Usamos el término "no excluyente", porque es claro que en toda negociación las posturas rígidas y unilaterales son inviables. Pero esta filosofía, que posibilita una visión globalizadora de los distintos intereses en juego, no debe conducir a que se renuncie a defender el interés propio sin ningún tipo de claudicaciones.

En este sentido, interpretando el mandato de la Asamblea General Extraordinaria, la conducción de CAESCO realizará un seguimiento puntual de todas aquellas alternativas que se presentan en las negociaciones que se llevan a cabo, y que en el caso del sector empresario son protagonizadas por los representantes de la Cámara Argentina de Comercio (CAC).

EXITO EN CORDOBA

Al cierre de esta edición toda la organización de CAESCO se movilizaba detrás de un objetivo central: hacer que el II Congreso de Empresas de Servicios Informáticos (COFESI), sea un éxito. Distintas delegaciones del interior del país ya habían confirmado su presencia en Córdoba, sede del encuentro, y un importante contingente de empresarios de Capital Federal se disponía a viajar hacia esa provincia.

Asimismo, distintas entidades públicas y privadas hicieron llegar su adhesión y su auspicio. Entre ellas cabe mencionar a la Secretaría de la Función Pública y la Subsecretaría de Sistemas de Información, ambos organismos dependientes de la Presidencia de la Nación; el Ministerio de Educación y Justicia, a través de la Secretaría de Justicia (resolución Nº 420); el Consejo Profesional de Ciencias Informáticas; el Consejo Federal de Informática; la Cámara Argentina de Comercio y la Asociación de Graduados en Sistemas de la Universidad Tecnológica Nacional.

BOLETIN DE LA CAMARA EMPRESARIA DE SERVICIOS DE COMPUTACION

Características técnicas del proyecto

El Entorno Operativo del proyecto fue el siguiente :

```
/*
 *PROYECTO : CRIBA DE ERATOSTENES
 *-Versión de Fecha : 20.4.88
 *-Compilador: LightSpeed'C' V. # 2.11 -Think Technologies, Inc.- (LSC)
 *-OPCIONES DE COMPILADOR A USAR :
 *    CHECK POINTER TYPES   Verifica Tipos de Apuntador
 *    CHECK PROTOTYPES      Verifica Prototipos de Funciones
 *    NOT CONFIRM SAVES     Guarda Modificaciones Automáticamente
 *-Entorno Operativo : APPLE MAC SE, 1MB RAM, 20 MB HARD DISK
 *-Descripción : Criba de Eratóstenes para hallar números primos
 *-Módulos a Incluir en Proyecto (ADD)- :
 *        SEGMENTO#1
 *                MacTraps(MAC Library)
 *                criba.c          (Versión A Elección)
 *                stdio            (LSC Library)
 *                strings          (LSC Library)
 *                unixtime.c       (Library Sources)
 *-Autor : Javier R. Blanqué © 1987, Derechos Reservados
 */
```

Glosario

Todo lenguaje tiene ciertos aspectos que pueden no gustarnos,... y que quisieramos cambiar, para eso construimos un glosario. Por ejemplo, yo tipeo a la velocidad de una tortuga manca, y prefiero lenguajes que sean lo más breves que sea posible, lo bueno es que en C uno puede casi definir su propio lenguaje, a través del Preprocesador, para lo cual creamos un archivo 'criba.h' donde incluimos las abreviaturas, y símbolos definidos por nosotros.

```
/*
 *criba.h :Librería Preprocesador (Glosario de símbolos/datos globales)
 */
#include <stdio.h>
/*       Abreviatura     Se expande a :       */
#define  UNS             unsigned
#define  REG             register
#define  EXIT            ExitToShell();
#define  CASE            break; case
#define  DEFAULT                  break; default
#define  EQ              ==
#define  NE              !=
#define  NOT             !
```

Ejemplo 1

Ahora viene el clásico ejemplo de cribe de Eratóstenes, usado en la mayoria de las revistas de computación para probar la velocidad de los equipos (Benchmarks) en aritmética entera, manejo de tablas, etc.

```
/*
 *criba.c : CRIBA DE ERATOSTENES Versión 1
 *        Criba de Eratóstenes Clásica
 *        Como se conoce en las Publicaciones
 *Prueba 1 :
 *        Tiempo Total de generación de tabla : 1,4 Segundos
 *        Tiempo de grabación : 12 Segundos
 *        Máximo Nº Natural procesado : 32767
 *        Cantidad de Primos Hallados : 3512
 *        Tamaño del Archivo generado : 19,5 KB
 */
#include "criba.h"
#define SIZE        32767    /* 2^15 - 1 */

/* Prototipos de las Funciones usadas :
        1 - Tipo que devuelve
        2 - Nombre de la Función
        3 - Tipos de los Parámetros
 */
int          main       (void);
UNS long     time       (unsigned long);
int          printf     (char *, ...);
int          fprintf    (FILE *,char *, ...);
FILE         *fopen     (char *,char *);

   main() {

/* Siempre que se pueda, debemos usar variables 'register' (REG), ya que
   se instalan en los registros de uso general de la CPU, y no se necesita
   transferirlos desde la memoria RAM, hay varios de ellos para datos
   variables y varios para direcciones (apuntadores) */

char FLAG[SIZE];  /* Criba de Números (si al final FLAG[i]=1 => es primo) */
REG   UNS    i, k;      /* Variables de trabajo */
REG   FILE   *f;        /* Archivo para guardar Números primos */
REG   UNS    long Tiempo;

printf("CRIBA DE ERATOSTENES\n");
/* Carga el tiempo Inicial */
Tiempo = time(0L);
/* Inicializamos Tabla -Criba- con unos */
for ( i = 0 ; i < SIZE ; i++)
        FLAG[i] = 1;
/* Si es primo pone 0 a todos sus múltiplos */
for ( i = 2 ; i < SIZE ; i++)
        if (FLAG[i] EQ 1)
                for ( k = i + i ; k < SIZE ; k += i)
                        FLAG[k] = 0;
/* Muestra el tiempo al final */
printf("\nTardó : %d Segundos\n", (int) (time(0L) - Tiempo));
/* Abre el Archivo para grabar los primos hallados */
f = fopen("Datos de Criba","w");
if (NOT f) {
        printf("ERROR EN apertura de archivo de salida");
        EXIT;
        }
/* Graba el Archivo con los primos hallados */
fprintf(f, "CRIBA DE ERATOSTENES\n");
for (i = 2 , k = 0 ; i < SIZE ; i++)
        if (FLAG[i] EQ 1)
                fprintf(f, "%d%s", i, ( (k++ % 10) ? ":" : "\n"));
fprintf(f, "\nCantidad de Números Primos : %d\n", k);
/*      No necesitamos cerrar el archivo,
        se cierra automáticamente al finalizar el programa */
/* fclose(f); */
}
```

Ejemplo 2

Lo más significativo de esta versión es el manejo de apuntadores en lugar de subíndices, y la posibilidad de direccionar a toda la memoria RAM.

```
/*
 *criba.c : CRIBA DE ERATOSTENES Versión 2
 *        Usa Apuntadores 'Pointers' para Optimización
 *Prueba 1 :
 *        Tiempo Total de generación de tabla : 0,8 Segundos
 *        Tiempo de grabación : 11 Segundos
 *        Máximo Nº Natural procesado : 32767
 *        Cantidad de Primos Hallados : 3512
 *        Tamaño del Archivo generado : 19,5 KB
 *Prueba 2 :
 *        Tiempo Total de generación de tabla : 26 Segundos
 *        Tiempo de grabación : 270 Segundos
 *        Máximo Nº Natural procesado : 815000
 *        Cantidad de Primos Hallados : 65052
 *        Tamaño del Archivo generado : 434 KB
 *Notas : (Sobre las Optimizaciones)
 *        Espacio :
 *                Se sortea el problema de la versión 1 usando la función
 *                del Sistema Operativo 'NewPtr', permite asignar bloques dinámicos
 *                de memoria hasta el total de RAM disponible (idem malloc en Unix)
 *                Permite Hallar Primos hasta el Nº 815.000 con (RAM de 1MB)
 *                teniendo en cuenta la memoria que ocupan el programa y el S/O
 *        Velocidad :
 *                El manejo a través de apuntadores permite direccionar
 *                directamente por sumas los vectores, y no por multiplicaciones
 */
#include "criba.h"
#define SIZE        815000L


/* Prototipos de las Funciones usadas :
        1 - Tipo que devuelve
        2 - Nombre de la Función
        3 - Tipos de los Parámetros
 */
int          main       (void);
UNS long     time       (unsigned long);
int          printf     (char *, ...);
int          fprintf    (FILE *,char *, ...);
FILE              *fopen     (char *,char *);

main() {
char    *FLAG;
REG char*PF, *Limite, *Paso;
REG long        j, Primo;
FILE            *f;
UNS long        Tiempo;


printf("CRIBA DE ERATOSTENES\n");
Tiempo = time(0L);
/* Asignación de memoria para tabla de elementos naturales hasta MAX RAM */
FLAG = (char *) NewPtr(SIZE * sizeof(char));
if (NOT FLAG) {
        printf("Sin memoria disponible\n");
```

```
		EXIT;
		}
	Limite = FLAG + SIZE;
	for ( PF = FLAG ; PF < Limite ; PF++)
		*PF = 1;
	for ( PF = FLAG + 2 ; PF < Limite ; PF++)
		if (*PF EQ 1) {
			Primo = PF - FLAG;
			for ( Paso = PF + Primo ; Paso < Limite ; Paso += Primo)
				*Paso = 0;
			}
	printf("\nTardó : %d Segundos\n", (int)(time(0L) - Tiempo));
	f = fopen("Datos de Criba","w");
	if (NOT f) {
		printf("ERROR EN apertura de archivo de salida");
		ExitToShell();
		}
	fprintf(f, "CRIBA DE ERATOSTENES\n");
	for ( PF = FLAG + 2, j = 0L ; PF < Limite ; PF++)
		if (*PF EQ 1)
	   fprintf(f, "%ld%s", (long)(PF - FLAG), ((j++ % 10L)?":":"\n"));
	fprintf(f, "\nCantidad de Números Primos : %ld\n", j);
	}
```

Ejemplo 3

En esta versión utilizamos la técnica de campos de bit para poder direccionar a las partes componentes de un byte. La tarea se hubiera simplificado mucho si la sintaxis del lenguaje permitiera vectores de bit, en este Release el compilador LSC no lo hace, (otros sí). Sería fantástico poder definir 'typedef struct TBLBIT { UNS bit[16] : 1; } TblBit;', para después preguntar por 'if (PF->bit[i] EQ 1)', pero cómo no se puede, lo hice tratando a cada uno de los 16 bits de la palabra cómo un caso especial (BAH!!).

```
/*
*criba.c :CRIBA DE ERATOSTENES Versión 3
*	Usa 'Bit Fields' que le permiten almacenar hasta 6,5 millones de marcas
*	para saber si un número natural es primo o no en un espacio de 810000B
*Prueba 1 :
*	Tiempo Total de generación sin salida por pantalla :  14 Segundos
*	Tiempo Total de generación con salida por pantalla :  117 Segundos
*	Tiempo de grabación :  14 Segundos
*	Máximo Nº Natural procesado : 32768
*	Cantidad de Primos Hallados : 3512
*	Tamaño del Archivo generado :  19,5 KB
*Prueba 2 :
*	Tiempo Total de generación de tabla (sin pantalla) :  344 Segundos
*	Tiempo de grabación :  288 Segundos
*	Máximo Nº Natural procesado : 815008
*	Cantidad de Primos Hallados : 65502
*	Tamaño del Archivo generado : 434 KB
*Prueba 3 :
*	Tiempo Total de generación de tabla (sin pantalla) : 2821 Segundos
*	Tiempo de grabación :  2244 Segundos
*	Máximo Nº Natural procesado : 6480000
*	Cantidad de Primos Hallados : 443481
*	Tamaño del Archivo generado : 3377,5 KB
*Notas :
*	Usa 'Bit Fields' para permitir hallar primos hasta el nº 6.480.000.
*	De acuerdo a las especificaciones de este compilador (LightSpeed C)
*	y de la mayoría de los que hay en uso en computadores personales,
*	los campos de bit no pueden ser vectores, por lo tanto no se puede
*	escribir algo como 'PF->Bit[i]', y la menor longitud direccionable
*	para estructuras que contienen campos de bit es de word o UNSIGNED
*	SHORT INT (16 bits), de modo que debemos tratar cada uno de los 16
*	bits de la estructura como un caso especial. Esto y el agregado de
*	las instrucciones en que se desglosa la manipulación de 'Bit Fields'
*	generan mayor cantidad de código -que es a la vez menos eficiente-.
*	Este es el principal factor en la diferencia de velocidad entre la
*	versión 3 (la más lenta), y las versiones 1 y 2 (la más rápida).
*	Sin embargo con esta versión obtenemos un mayor espacio direccionable
*	con la misma memoria RAM, un mejor aprovechamiento (alrededor de 8
*	veces más eficiente).
*	El procedimiento que se realiza para hacer rendir a un millón de
*	caracteres 6,5 M de marcas, es dividir una tabla de m elementos
*	donde m = n * k, siendo n la longitud real de la tabla de estructuras
*	y k es la longitud en marcas SI / NO - 1 / 0 - de cada estructura.
*/
#include "criba.h"


/*
SIZE : Cantidad de estructuras * 16 = Cantidad de Marcas
	Prueba 1 - 2048 * 16 = 32768
	Prueba 2 - 50938 * 16 = 815008
	Prueba 3 - 405000 * 16 = 6480000
*/
#define SIZE	405000L /* Cantidad de Palabras(c/u tiene 2 Bytes) */
```

```
/* Prototipos de las Funciones usadas :
	1 - Tipo que devuelve
	2 - Nombre de la Función
	3 - Tipos de los Parámetros
*/
int		main		(void);
UNS long	time		(unsigned long);
int			printf		(char *, ...);
int			fprintf		(FILE *,char *, ...);
FILE			*fopen		(char *,char *);
void			gotoxy		(int, int);

/* Estructura o Registro que divide una palabra (WORD) en 16 Bits */
typedef struct TBLBIT {
	UNS bit0 : 1, bit1 : 1, bit2 : 1, bit3 : 1;
	UNS bit4 : 1, bit5 : 1, bit6 : 1, bit7 : 1;
	UNS bit8 : 1, bit9 : 1, bitA : 1, bitB : 1;
	UNS bitC : 1, bitD : 1, bitE : 1, bitF : 1;
	} TblBit;

main() {  /*Programa Principal */
FILE	*f;
TblBit	*FLAG;
register	TblBit *PF;
register	long i, j, k;
long	PRIME, LIMITE, PASO, TEMP, x;
UNS long		Tiempo;

Tiempo = time(0L);
gotoxy(30, 10);
printf("CRIBA DE ERATOSTENES\n");
gotoxy(50, 18);
printf("PRIMO HALLADO : ");
FLAG = (TblBit *) NewPtr(SIZE * sizeof(TblBit));
if (NOT FLAG) {
	printf("\nSin Memoria.\nElementos %ldUnidad %d\n",SIZE,sizeof(TblBit));
	EXIT;
	}
/* Multiplico por 16 (Shift a izq. 4 bits) más rápido */
LIMITE = SIZE << 4L;
for ( i = 0L ; i < LIMITE ; i += 16L) {
	k = i >> 4L;		/* Divido por 16 (Shift a der. 4 bits) más rápido */
	for( j = 0L ; j < 16L ; j++) {
	/* Tomo dirección de elemento k en apuntador para más velocidad */
		PF = &FLAG[k];
		switch(j) {
			case,0: PF->bit0 = 1;
			CASE 1: PF->bit1 = 1;
			/* CASE 2 HASTA CASE 14: PF->bit 2 HASTA 14 = 1; */
			CASE 15: PF->bitF = 1;
			DEFAULT: printf("Error : no existe bit nº : %ld\n", j);
			}
		}
	}
for ( i = 2L ; i < LIMITE ; i++) {
	k = i >> 4L;
	j = i - (k << 4L);
	PF = &FLAG[k];
	switch(j) {
		case 0: if (PF->bit0 EQ 1) goto Seteo;
		CASE 1: if (PF->bit1 EQ 1) goto Seteo;
	/* CASE 2 HASTA CASE 14:  if (PF->bit 2 HASTA 14 EQ 1) goto Seteo; */
		CASE 15: if (PF->bitF EQ 1) goto Seteo;
		DEFAULT: printf("Error : no existe bit nº : %ld\n", j);
		}
	continue;

/*	La programación BIEN estructurada nos indica que podemos usar
	la sentencia GOTO siempre que implique claridad y / o ayude a
	la eficiencia de administración de espacio o tiempo. */
Seteo:
/* Para ver por pantalla c/primo sacar comentario a próxima linea */
/*	gotoxy(70, 18); printf("%ld", i); */
	for ( PASO = i + i ; PASO < LIMITE ; PASO += i) {
		TEMP = PASO >> 4L;
		x = PASO - (TEMP << 4L);
		/* Tomo la dirección del elemento para mayor velocidad */
		PF = &FLAG[TEMP];
		switch(x) {
			case 0: PF->bit0 = 0;
			CASE 1: PF->bit1 = 0;
			/* CASE 2 HASTA CASE 14: PF->bit 2 HASTA 14 = 0; */
			CASE 15: PF->bitF = 0;
			DEFAULT: printf("Error : no existe bit nº : %ld\n", x);
			}
		}
	}
printf("\nTiempo de Generación : %lu\n", (time(0L) - Tiempo));
Tiempo = time(0L);
f = fopen("Datos de Criba","w");
if (NOT f) {
```

```
        printf("ERROR EN apertura de archivo de salida");
        EXIT;
        }
  fprintf(f, "CRIBA DE ERATOSTENES\n");
  for ( i = 2L, TEMP = 0L ; i < LIMITE ; i++) {
        k = i >> 4L;
        j = i - (k << 4L);
        PF = &FLAG[k];
        switch(j) {
                case 0: if (PF->bit0 EQ 1) goto Grabo;
                CASE 1: if (PF->bit1 EQ 1) goto Grabo;
        /* CASE 2 HASTA CASE 14:  if (PF->bit 2 HASTA 14 EQ 1) goto Grabo; */
                CASE 15: if (PF->bitF EQ 1) goto Grabo;
                DEFAULT: printf("Error : no existe bit nº : %ld\n", j);
                }
        continue;
      Grabo:
              fprintf(f, "%ld%s", i, ((TEMP++ % 10L)?":":"\n"));
              }
      fprintf(f, "\nCantidad de Primos : %ld\n", TEMP);
      printf("\nTiempo de Grabación : %lu\n", (time(0L) - Tiempo));
      }
```

Parte del Archivo Generado

El archivo ocupaba 3,5 Megabytes en disco, si fuera necesario almacenarlo en forma permantente, para proyectos como encriptación de claves, etc., puede reducirse el espacio ocupado mediante la grabación binaria (3 o 4 bytes por Nº), o mejor, la grabación de la tabla de marcas, con lo que se reduciría el natural a su ordinal con un SI/NO para saber si es primo.

```
2:3:5:7:11:13:17:19:23:29:31
37:41:43:47:53:59:61:67:71:73
79:83:89:97:101:103:107:109:113:127
131:137:139:149:151:157:163:167:173:179
881:883:887:907:911:919:929:937:941:947
953:967:971:977:983:991:997:1009:1013:1019...
9907:9923:9929:9931:9941:9949:9967:9973:10007:10009...
32749:32771:32779:32783:32789:32797:32801:32803:32831:32833...
65521:65537:65539:65543:65551:65557:65563:65579:65581:65587...
99991:100003:100019:100043:100049:100057:100069:100103:100109:100129...
999917:999931:999953:999959:999961:999979:999983:1000003:1000033:1000037...
1999979:1999993:2000003:2000029:2000039:2000081:2000083:2000093:2000107:2000113...
2999921:2999933:2999951:2999957:2999999:3000017:3000029:3000047:3000061:3000073...
3999917:3999923:3999929:3999949:3999971:4000037:4000039:4000043:4000063:4000067...
4999963:4999999:5000011:5000077:5000081:5000087:5000101:5000111:5000113:5000153...
5999909:5999911:5999921:5999923:5999927:5999933:5999947:5999993:6000011:6000023...
6478673:6478723:6478729:6478733:6478757:6478763:6478777:6478781:6478793:6478831...
6479663:6479701:6479729:6479743:6479747:6479791:6479801:6479807:6479827:6479831
6479897:6479903:6479917:6479923:6479929:6479933:6479953:6479959:6479989:6479999
```

Conclusión

El tema más importante en el lenguaje C no corresponde al lenguaje en sí, sino a si tenemos en claro el problema, lo que hace C es ayudarnos a pensar en un sentido estructurado y oscurecer funciones -la metáfora de la Caja Negra-. C no es perfecto, hay múltiples versiones, unas mejores que otras, pero puede crecer, tanto por parte de los usuarios, como de los fabricantes y productores de compiladores, y ANSI C es un indicio. La mayor parte de productos de software importantes desde Unix a Quatro, o desde dBase IV a Eureka, estan hechos en C. Actualmente hay extensiones para trabajar con bases de datos tipo SQL o de red, interfases gráficas, sistemas expertos en inteligencia artificial. Y C++, un lenguaje C extendido, orientado a objetos (OOL), de mayor nivel que el C (10).

C es un lenguaje de uso genérico, no le pidamos que haga en determinado campo lo que hacen lenguajes diseñados específicamente para trabajar con ese ambiente. Sin embargo el usuario puede escribir sus métodos en C y lograr un lenguaje orientado equivalente de performance superior. De esta forma se desarrollan casi siempre los lenguajes específicos. Para muestra basta un botón : MACSYMA es un paquete escrito en LISP -Intérprete que a su vez generalmente está escrito en C- en el MIT (Instituto Tecnológico de Massachusetts) que permite manipular expresiones algebraicas, y reducir su complejidad automáticamente. Hace unos años el proyecto fue renombrado como SMP (Symbolic Manipulation Program), extendido y traducido a C por el Cal-Tech (Instituto Tecnológico de California), consta de 90 mil líneas de código fuente, y es diez veces más rápido que MACSYMA (11).

Luego de haber trabajado con más de una docena de lenguajes, y haber formado parte de muchos proyectos interesantes tanto en Mainframes como Minis y PCs, llegué a conclusión de que la única forma de aprender a querer -y odiar a veces- a un lenguaje -C en este caso- es trabajar con él, y tener en cuenta que lo que hagamos no será mejor que lo que nuestra capacidad y esfuerzo permitieron -independientemente del lenguaje-, el C NO nos dará soluciones mágicas, aunque a veces aparezcan 'viejos trucos'. La magia está en el País de las Maravillas y en los niños.

Referencias

1 • Desarrollo de Software de Base : Una Experiencia
Javier Blanqué, Mauricio Fernandez, Lisandro Ortiz y Alberto Seijas
Mundo Informático, 2ª Quincena de Febrero de 1987, Argentina

2 • El Lenguaje C
Ricardo Forno
Mundo Informático, Febrero de 1988, Argentina

3 • Ponderación del Lenguaje C
Eduardo Losoviz
Mundo Informático, 1ª Quincena de Marzo de 1988, Argentina

4 • 1987-1988 : Lo que pasó y lo que vendrá o cómo compactar un lustro en un año
Javier Blanqué
Mundo Informático, 2ª Quincena de Diciembre de 1987, Argentina

5 • Preparing for ANSI C
Richard Relph
Software Tools, August 1987, U-SA

6 • Optimizing Compilers
Mark Roberts
Byte, October 1987, USA

7 • Soportes para desarrollo de sistemas expertos basados en lógica difusa
Javier Blanqué
IV Jornadas de Inteligencia Artificial y Robótica, Univ.Belgrano, 1987, Argentina

8 • Notas de Algebra I
Enzo R. Gentile
EUDEBA, Ediciones Previas, 1973, Argentina

9 • An Introduction to the Theory of Numbers
G.H.Hardy y E.M.Wright
1954, USA

10 • Lenguajes orientados a conceptos
Javier Blanqué
Mundo Informático, 2ª Quincena de Julio de 1987, Argentina

11 • El Sistema Unix y sus Aplicaciones
Jose Canosa
Marcombo, 1984, España

Asociacion Argentina de Teoria General de Sistemas y Cibernética

Escribe: Charles Francois

LA ORGANIZACION-SISTEMA EN SU ENTORNO

Ninguna empresa u organización está "sola en el mundo". Al contrario, cumple con una serie de funciones que tienen sentido sólo si dan en relación con el mundo circundante.

No percibir esta situación constituye la causa principal de los múltiples problemas que suelen acosar a los decisores. Si tienen conciencia clara de su dependencia más o menos grande respecto de su entorno, pueden llegar a un cierto grado de autonomía, o sea, de capacidad propia de previsión y de acción, en función de las variaciones externas.

Las organizaciones-sistema son claramente distintas de su entorno, pero viven con él en necesaria simbiosis. Es destacable que sólo la parte material fija de la organización es permanentemente identificable como su infraestructura. Todo el resto, desde los flujos de energía, materia o información, hasta los propios colaboradores de la organización, presentan una especie de ambigüedad participativa. La energía se transforma y se disipa, las materias primas sirven para la elaboración de productos destinados al entorno, la información es un flujo que se modifica constantemente en función de las circunstancias exteriores y también tiene sentido sólo en función de sus intérpretes. Estos últimos se ponen a disposición de la organización pero sólo en algunos aspectos, bajo algunas condiciones y por un tiempo. Conservan sus propias motivaciones en función de valores, normas y metas que están lejos de coincidir totalmente con las de la organización, y cuyas interpretaciones varían con el tiempo.

La evolución del concepto de empresa u organización, desde el siglo XIX, es muy notable en este sentido. Los "Capitanes de Industria" crearon la gran empresa moderna en un marco de coincidencia entre sus ambiciones mercantiles y su genio técnico, en circunstancias históricas favorables al desarrollo del capitalismo. En general, usaron hábilmente las condiciones de entorno que se les presentaron. Pero, por su propia acción, las modificaron en profundidad, sin advertir claramente los cambios.

Es así como, sucesivamente, el entorno capitalista reaccionó con la aparición del sindicalismo, del anti-colonialismo, de la concentración oligopólica, del consumismo masivo y los controles (muchas veces monopólicos del Estado). La transformación, lejos de estabilizarse, no hace más que acelerarse. Ahora presenciamos las mundialización de los mercado financieros y de productos; la reducción del rol de los productores primarios y secundarios a favor de los dadores de servicios; el auge de la mega-ciencia cada vez más costosa; la aceleración de una evolución tecnológica en muchos casos netamente desestabilizadora.

Ya ningún decisor puede prescindir de una observación atenta y constante sobre el entorno de la organización a su cargo, basada en una información idónea, so pena de correr grandes riesgos.



Escribe: Rubén Lodeiro

TECNOLOGIA DE PUNTA: AS 400 DE IBM

En nuestro encuentro anterior comprometí esta entrega a comentar el impactante anuncio realizado por IBM en el pasado mes de junio; el nuevo equipo IBM AS/400. Como lo prometido es deuda, nada mejor que entrar en tema.

Se trata de una nueva serie que reemplaza a la "vieja" línea S/3X y que, junto a la S/9370 pasan a conformar las columnas de comercialización de "mainframes" de la principal empresa proveedora de hardware de nuestro mercado.

Características técnicas del hardware:

Fueron anunciados seis modelos, los iniciales (B10 y B20) son compactos y vienen provistos de una unidad de proceso 9404, en tanto que los mayores (B30, B40 y B60) se instalan montados en "racks" y ofrecen una CPU 9406.

A modo de resumen se mencionan las características principales de cada uno de los modelos.

1. Unidad central de procesamiento: (CPU)

	9404		9406			
	B10	B20	B30	B40	B50	B60
Velocidad (en nanosegundos)	120	120	60	60	60	60
Rendimiento relativo (B10=1,0)	1,0	1,3	1,5	2,3	3,5	5,5
Máximo de terminales ASCII	36	36	vía	conv.	de	protocolo
Máximo de terminales TWINAXIALES	40	40	120	200	320	480
Memoria principal (en MB)	4/8	4/16	4/36	8/40	16/48	32/96
Megainstrucciones por segundo	0,25	0,40	0,50	0,75	1,15	2,00
Canales de datos E/S	1	1	1	1	2	3
Máximo de almacen. en discos (GB)	0,945	0,945	6,8	6,8	13,6	27,2
Máximo de unidades de diskette	1	1	2	2	2	2
Máximo de líneas de comunicaciones	6	6	16	32	32	32
Máximo de conexiones LAN-Token Ring	1	1	2	2	2	2

2. PERIFERICOS

2.1 Unidades de disco:

Se anuncian tres modelos cuyas características son las siguientes:

	Capac. (Mb)	Veloc Transf. (Mb/s)	Demora Prom. (ms)
6100	315	1,25	23,0
9332	200/400	2,50	19,5
9335	850	3,00	18,5

2.2 Unidades de Cinta Magnética:

Los modelos compatibles son 9346, 9347 y 2440.

Los controladores 2601 y 2602 soportan un dispositivo 9346/7 y un 2440, respectivamente.

A partir del Release 1.2 también serán compatibles los modelos 3422 de 780 Kb/s y 3430 de 312 Kb/s.

2.3. Unidades de diskettes:

Los modelos anunciados son el 933-MI (8", 1, 2Mb) y 9331-M2 (5,25", 1, 2Mb). No existe soporte para magazine en la línea AS/400.

2.4. Estaciones de trabajo:

Son compatibles todas las disponibles hoy para línea S/3X, o sea:

TWINAXIALES:3179-002, 3180-002,3196 A10/A20/B10/B20,3197-C10/C20/D10/D20,3197-W10/W20 (sólo mediante dispositivo 6040), 5251-011/999, 5291-001/ 32 y 5292-001.

Además se soportan las modelo 3270 vía LPC-5209 (link Protocol Converter) o emulador 3270.

ASCII:" 3101, 3151, 3161, 3163 y 3164 vía LPC-5208 en modelos B30 en adelante y mediante conexión directa en B10 y B20.

2.5 Controladores:

5251-M12 (8 dispositivos), 5294 (8 dispositivos) aunque con dificultad en transferencia para PCs remotas y 5394 (16 dispositivos) aún no disponible en Argentina que soluciona el problema anteriormente mencionado.

2.6. Impresoras

3812-001/002,5219-DO1/DO2,5224-001/002, 5225/001/002/003/0045256-001/002/003,5262-001,6262-T12/T14,4210-001,4224-101/102/1C2/1E2,4234-002, 4245-T12/T20 y 4214-002 y tipo ASCII 4201-002, 4202-001, 4207-001, 4208-001, 4224-301/302 y 5204-001.

3. Comunicaciones:

Soporta los siguientes protocolos y redes:

— Lan-Tokeng Ring (IEEE 802.2 y 802.5)
— X.21
— X.25
— Asincrónicas (ASYNC)
— Sincrónicas (BSC)
— Synchronous Data Link Control (SDLC)

Ofrece la facilidad de ECS (Electronic Customer Support), que consiste en la posibilidad de acceder on-line al servicio de mantenimiento IBM y a sus Bases de Datos disponibles tanto localmente cuanto en el exterior. Opera una vez finalizado el IMPL.

4. Configuración:

Aún no está disponible el configurador AS/400. Es importante una correcta definición del Rack ya que no es fácil reconfigurarlo.

Características del Software:

El AS/400 está provisto del sistema operativo OS/400 capaz de soportar tres diferentes ambientes de trabajo:

— Nativo (AS/400)
— S/36
— S/38

Es factible también correr aplicaciones con mezcla de entornos. Los lenguajes disponibles son BASIC, CLL, COBOl 74 y 85, FORTRAN, RPGII, RPGIII, RPG400, PASCAL y PL/I.

El OS/400 es capaz de administrar bases de datos relacionales explotables vía lenguaje de programación y/o SQL y un sistema de seguridad de alta sofisticación. Asimismo, se han respetado los lineamientos de la SAA (Systems Application Architecture).

Esta línea ofrece también, una completa colección de material de aprendizaje compuesto de tres niveles:

— El primero, que comprende conceptos generales, terminología y conocimientos mínimos de facilidades y operación es el denominado TSS (Tutorial System Support).

— Si la necesidad de capacitación es más puntual y profunda, se aconseja utilizar las Education Series y, finalmente

— Existe una tercera opción llamada SEF (Self Education Facility), que permite a los instructores generar sus propios capítulos de enseñanza.

Por supuesto que existen infinidad de tópicos interesantes como para extenderse en el análisis, pero, por razones de espacio me limité a generar una breve síntesis del frondoso informe que gentilmente me brindaron el Lic. Carlos Anino Campana (Interactiva S.A.) y el C.C. Horacio Jezzi Riglos (CGA-Sistemas de Cía. Gillette de Argentina S.A.) a quienes agradezco su desinteresada colaboración.

Estos mismo profesionales se han comprometido a enviarme un reporte de sus experiencias en migración de aplicaciones desde S/36 y S/38, aporte que desde ya publicaremos en esta columna.

Queda entonces para una futura oportunidad el tratamiento de la propuesta IBM de transportabilidad de aplicaciones desde PC, AS/400 y S/370 llamada SAA (Systems Application Architecture).

¡Hasta la próxima!



El rincón del usuario de la PC

Escribe: Jorge Gottheil

CAPACITACION EN PC, COMO COMENZAR, QUE HACER PRIMERO.

Una de las preguntas que más frecuentemente se me formulan, es Dónde puedo hacer un buen curso para Lotus.?, o Donde puedo hacer un curso sobre PC?

No importa la pregunta especifica, las mismas implican una total aceptación de que con la compra de una PC no se soluciona nada, hay que operarla y algo más.

Generalmente transcurre cierto tiempo para que se acepte este hecho, dependiendo de la cultura informática de cada interlocutor.

En mi labor profesional, la de desarrollar sistemas y capacitación, me solicitan muchas aplicaciones sencillas de resolver para mí con un desarrollo simple en Lotus, y si este es el caso, aconsejo que se capaciten 2 a 3 personas de la empresa, finanzas, compras y/o costos, en el manejo del Lotus, y durante el dictado del curso vamos desarrollando estas aplicaciones, que luego son mantenidas y ampliadas por los mismos capacitados.

Como filosofía general, es importante comenzar la capacitación con una introducción de dos a tres hs. sobre la PC, su configuración, algo de historia, administración de la memoria del equipo, que programas la usan y cuánto, etc

Luego la elección sobre qué soft de base capacitar está muy ligada a la actividad, pero por regla general, es casi seguro que saber Lotus 1 2 3 no va a estar de más en casi ninguna organización. Esto es así pues siempre se deben realizar planillas de algún tipo para llevar algún tipo de control, y esto se puede hacer en Lotus 1 2 3.

Una de las dificultades mayores de los asistentes a cursos, es imaginarse qué pueden hacer en la planilla electrónica, es decir, les cuesta aplicar lo aprendido a los casos prácticos.

Por esta razón los cursos multitudinarios donde no se desarrollen ejemplos específicos para cada participante, resultan a veces poco rendidores y decepcionantes, ya que se obtienen un montón de conocimientos de comandos totalmente extraños a la cultura normal, y no se tiene claro para qué sirven.

Otra función importante del asistente a un curso, es practicar mucho sobre ejemplos reales, lo ideal sería practicar entre clase y clase, el doble de las horas de curso de la clase anterior.

La frecuencia de las clases es importante. No soy partidario de los cursos tipo seminario, ya que el torrente de conocimientos a absorber, sumado al poco o casi nulo tiempo de práctica, los hace poco efectivos.

La frecuencia ideal a mi entender, es de unas cuatro horas, por semana en dos veces o tres horas. por semana en una clase. La elección depende del tiempo disponible para practicar de los asistentes. Y vuelvo a hacer hincapié en esto último, ya que la práctica genera dudas, va incentivando la imaginación, y luego vienen las preguntas, Cómo hago tal cosa?, Me pasó tal otra, etc.

Generalmente el que no hace preguntas en la tercer clase, es que no ha practicado lo suficiente

Con respecto a la cantidad de clases, suelo reservar dos horas para usarlos, a los 15-30 días de finalizado formalmente el curso, para aclarar dudas y preguntas que surgen en la aplicación de casos específicos.

Un consideración importante es el horario, las primeras horas de la mañana son las más propicias, y si parte de esas horas se deben tomar del tiempo libre de los empleados, habrá una mejor valoración del curso considerando que es una capacitación personal, útil en caso de abandonar la empresa. Si es politicamente viable, lo justo sería dar las clases la mitad del tiempo en el horario de trabajo y la otra en el tiempo libre.

Nunca dejo que participen más de tres alumnos por equipo, lo ideal es dos. Si no se cuenta con suficientes equipos es necesario hacer varios turnos.

Los cursos introductorios de Lotus 1 2 3 con las características mencionadas, se pueden dictar en 12-15 horas dependiendo de la combinación de los factores citados. Una segunda etapa que abarca macroinstrucciones, gráficos avanzados, base de datos avanzada, etc. se puede dar en 15-20 horas.

Bueno, esto es todo por ahora, y recuerden que si desean consultar algún tema específico de las PC, dirigirse a Mundo Informático, Rincón del Usuario de la PC

POR LAS EMPRESAS

TEXAS INSTRUMENTS

TEXAS INSTRUMENTS TI 955 AT COMPATIBLE

Texas Instruments incorpora un nuevo modelo a su línea de computadoras personales, el TI 955. Diseñado para ser compatible con el standard del mercado PC-AT puede funcionar a una velocidad de 12 Mhz, brindando una altísima performance de proceso. La versatilidad del sistema lo hace apto para funcionar como estación de trabajo inteligente corriendo aplicaciones de MS-DOS o a través de su software de emulación conectarse a un ambiente de procesamiento multiusuario.

El TI 955 utiliza el microprocesador de 16 bits 80286 para una alta perfomance de procesamiento. La velocidad de trabajo del mismo puede ser seleccionable a 12 Mhz, o para aplicaciones que así lo requieran puede funcionar a 8 Mhz. El TI 955 posee 1 Megabyte de memoria RAM de 120 nanosegundos de acceso en la placa de la unidad del sistema, expandibles a 4 Megabytres utilizando una ranura de expansión. La unidad de diskettes de 1.2 Mb es standard pudiendo leer también el formato de 360 Kb. Una unidad de diskettes de 1.2 adicional o un disco rígido de 50 Megabytes son opcionales que pueden agregarse. El TI 955 fue diseñado para correr la infinidad de aplicaciones escritas bajo MS-DOS para PC AT. Con el software de emulación TI 924/931 es posible conectarlo como una terminal en un ambiente multiusuario.

TEXAS INSTRUMENTS TI OMNILASER SERIE 2000

Las nuevas Impresoras Laser de Texas Instruments están desarrolladas con la tecnología de segunda generación.

La gran capacidad de proceso, su alto ciclo de trabajo y la vida útil, sumados a una gran versatilidad en el manejo de papel y su gran calidad de impresión, hacen de éstas la elección más confiable en impresoras de página Laser de alta resolución.

Los cuatro diferentes modelos se adaptan a cualquier configuración y requerimientos de impresión, tanto para procesamiento de datos, procesamiento de palabra y autoedición.

Modelo 2015

Impresora de alto ciclo de trabajo y prolongada vida útil, su alta velocidad de impresión, — 15 páginas por minuto— la hacen ideal para aplicaciones de procesamiento de datos, donde existan requerimientos de gran velocidad de impresión (comparables con una impresora de línea de (900 LPM) y alta calidad de salida.

Las diferentes emulaciones de escape y control y sus variadas formas de conexión permiten su utilización en ambientes de multiprocesamiento o redes locales de trabajo.

Características

- Velocidad : 15 PPM
- Vida útil: 1,5 millones de copias
- Ciclo de trabajo: 25.000 copias mensuales
- Memoria RAM: 512 KB
- Emulación: TI855, Diablo 630, HP Laserset.
- Intefaces: Serie RS-232-C, Paralela Centronics.
- Font: Courier, Courier Oblique(*)
- Alimentación: 2 bandejas de 250 hojas cada una
- Tamaño de hoja: Carta, oficio y A4
- Resolución: 300 x 300 DPI a 1/2 pág.; 150 x 150 DPI a página completa.
- Gráficos: 300 x 300 DPI a 1/2 pág.; 150 x 150 DPI a página completa

(*) Hay otras tipografías disponibles en cartridges opcionales

Modelos 2106, 2108, 2115

Tres impresoras Laser con todo el poder del Postcript, su gran capacidad de memoria 2 y 3 Mb RAM y un poderoso microprocesador de 32 Bits ofrecen la posibilidad de realizar cualquier tipo de composición gráfica a una resolución de 300 x 300 puntos por pulgada. Sus fonts de tipografías y estilos residentes permiten la combinación de hasta 35 estilos junto a dibujos y fotografías (medios tonos) en una misma hoja. La posibilidad de agregar más tipografías en memoria o en cartridges opcionales expanden aún más las posibilidades de composición. La calidad de emulación permite utilizarla con cualquier software. Sus múltiples interfaces posibilitan la conexión con cualquier tipo de computación y son compatibles incluso con las incompatibles. Soportan distintos tamaños y tipos de papel, desde sobre y etiquetas, hasta acetato transparente.

IBM: NUEVOS MODELOS QUE AMPLIAN LA FAMILIA DEL SISTEMA PERSONAL/2

IBM acaba de ampliar su familia de productos del Sistema/2 (PS/2) con nuevas unidades de mesa que incluyen el sistema de cómputos más rápido que la empresa haya diseñado hasta el momento.

Los nuevos modelos 70-386 y 50 Z poseen memoria de alta velocidad y están equipados con la arquitectura Microcanal (MR) de IBM, que mejora la integridad de los datos y permite a los usuarios ejecutar programas extensos y múltiples con mayor seguridad y eficiencia.

También brinda compatibilidad entre el Sistema Operativo en Disco (DOS) y el Sistema Operativo/2 (OS/2), provee un subsistema avanzado de Matriz para Gráficos de video y bocas de conexión en serie, en paralelo y para dispositivos de señalamiento integradas en el panel de sistema, todos ellos dispositivos comunes en las máquinas PS/2 del nivel superior (basadas en Microcanal).

Ambos modelos poseen paneles de sistema más pequeños y de nuevo diseño que mejoran la calidad y la confiabilidad.

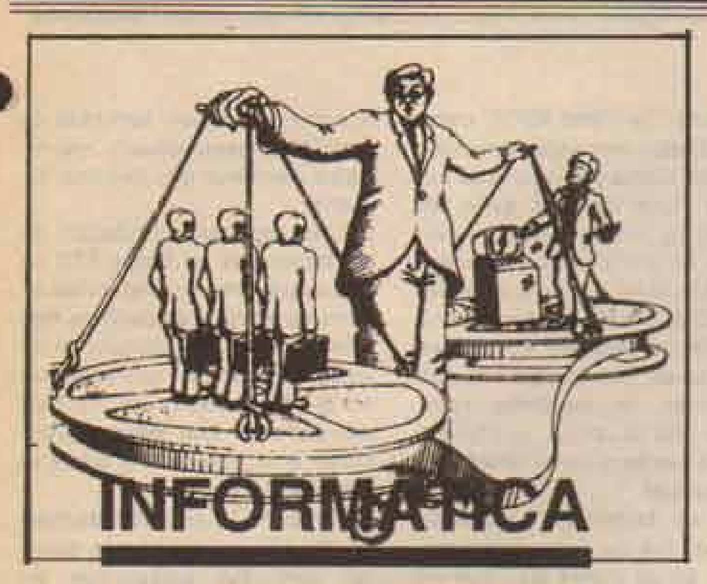
MEJORAS PARA APLICACIONES EDITORIALES

Se encuentran disponibles ahora nuevas versiones del programa bajo licencia Adoptador para Impresoras de Páginas de Computadoras Personales IBM, del programa diagramador de páginas Aldus PageMaker (MR) y del programa de manejo del entorno de sistema Windows Microsoft (MR), para su inclusión en los ofrecimientos de Solution Pac (MR) de IBM para aplicaciones editoriales personales.

El programa bajo licencia del adaptador contiene el lenguaje de descripción de páginas de Adobe Postcript (MR).

Los usuarios también pueden acceder al Subsistema de Impresora de Páginas de computadoras Personales de IBM para aplicaciones de impresión de gran calidad mediante láser, adaptadas a configuraciones específicas para trabajos editoriales y de otro tipo.

En aplicaciones basadas en un sistema central, un nuevo programa IBM bajo licencia, el Traductor de Lenguaje de Marcación Generalizado Standard (SGML) para composición DCF (Recurso de Composición de Documentos), permite a los usuarios cumplir con la norma internacional SGML para la preparación electrónica de documentos, al utilizar computadoras de grandes sistemas IBM



Y DERECHO

Escribe: Antonio Millé

UNA NUEVA MESA REDONDA INTERNACIONAL EN BUENOS AIRES REUNIRA A INFORMATICOS Y JURISTAS

Las "Mesas Redondas Internacionales Interdisciplinarias" constituyen encuentros anuales entre profesionales de la Informática y del Derecho, que el INSTITUTO LATINOAMERICANO DE ALTATECNOLOGIAINFORMATICA Y DERECHO (ILATID) viene organizando en cooperación con la Cámara de Empresas de Software (CES) y con el apoyo de entes oficiales nacionales y de organismos internacionales desde hace dos años.

La primera reunión, realizada en octubre de 1986, tuvo por tema "Software y Derecho de Autor" y contó con la participación del especialista norteamericano Henry Keplinger, de los profesores franceses André Francon y André Bertrand, del científico italiano Antonio Martino, de los profesores Antonio Chaves, de Brasil y Fernando Jordan Flores, de Colombia, junto con un nutrido equipo de profesionales argentinos de las dos ramas concernidas por la convocatoria.

La reunión de 1987 se dedicó al tema "Bancos de Datos y Propiedad Intelectual", siendo presidida por el eminentísimo catedrático francés Pierre Catalá, uno de los pioneros de la aplicación de la informática a la información jurídica.

La reunión correspondiente a 1988 tendrá lugar del 19 al 21 de octubre, dentro del 2º Congreso Iberoamericano de Software (CIBSO) y cuenta con la especial cooperación del Centre International de Recherches et d'Etudes du Droit de l'Informatique et des Telecomunications (CIREDIT), de Francia. Al igual que en las ocasiones anteriores, la Mesa redonda está patrocinada por la Secretaría de Justicia y cuenta la declaración de Interés Nacional.

En esta, su tercera participación, La Mesa Redonda Internacional Interdisciplinaria tendrá como título "Comercialización y Licenciamiento de Software", dedicándose a considerar temas relacionados con la comercialización, la protección y el licenciamiento del software en los distintos países del mundo, en un intento de brindar información actualizada y de alta especialización acerca del marco fáctico y reglamentario que rodea la importación y exportación de licencias para la comercialización de software en las distintas áreas regionales del mundo.

Los organizadores han dedicado una jornada de cada uno de los tres días que dura la reunión, al estudio de los problemas relacionados con una región del mundo. El primer día se estudiará lo relativo a América del Norte y Japón, el segundo día estará destinado a los países que forman la Comunidad Económica Europea y el tercer día a los países latinoamericanos.

En cada ocasión, durante las sesiones matutinas especialistas en la gestión de ventas de primer nivel internacional, expondrán al público las características especiales que asume la comercialización del software en uno de los mercados regionales, aludiendo a la estructura de la cadena de distribución, las políticas de marketing habituales, los precios y comisiones usuales, las condiciones corrientes en los negocios y consejos de orden práctico para encarar con éxito una operación de importación o exportación de licencias.

En las sesiones de las tardes, expertos legales de distintos países detallarán cómo está constituida la protección legal del software en el área respectiva, los recaudos que deben cumplirse para efectivizar tal protección, las formalidades que se requieren a una empresa extranjera para actuar dentro del país, las cláusulas contractuales que resultan normales para la negociación del software, los impuestos que se deben pagar en este comercio, los procedimientos para la remesa de regalías y en general consejos de orden práctico para labrar contratos eficaces y exigibles.

En todos los casos, se tratará de dar un enfoque rigurosamente práctico a las cuestiones y se brindará al público amplia oportunidad de efectuar consultas y de expresar sus inquietudes a los disertantes y al resto de los participantes.

Manteniendo el espíritu manifestado en años anteriores, la Mesa Redonda brindará a nuestros profesionales de la informática y del derecho la oportunidad de compartir jornadas de intensa y productiva actividad intelectual con especialistas que ocupan el primer rango mundial en la actividad jurídica dedicada a los problemas informáticos. Entre estos visitantes se hallarán:

— Phillippe Le Clech. Jurista especializado en comercialización internacional de software. Consejero Jurídico Principal de SLIGOS, Redactor Jefe de "DI-SEP - La Revue du Droit International des Systémes Electroniques de Paiement".

— Michael D. Scott. Jurista e Ingeniero de sistemas. Director de las revistas "Software Protection" e "International Computer Legal Adviser". Autor de numerosos trabajos e importantes libros sobre la materia.

— Marina Cousté. Jurista. Especializada en aspectos referentes a la protección del software en la Comunidad Económica Europea.

— André Bertrand. Jurista. Consejero Legal Principal para el Area Europa de NCR. Presidente de la Association Francaise pour le Droit de l'Informatique. Presidente del Centre International de Recherches et d'Etudes du Droit de l'Informatique et des Telecomunications (CIREDIT).

A este elenco de disertantes que ya ha confirmado su participación, se sumará en el futuro próximo un sólido equipo latinoamericano, que dará a conocer los problemas de nuestra región a los numerosos inscriptos extranjeros que se esperan para la reunión.

LA BATALLA DEL CHIP: NOVEDADES DEL CORRESPONSAL DE GUERRA

Otra vez la poderosa Semiconductor Industry Association, que nuclea a los productores de Silicon Valley, presiona a su gobierno para que haga efectiva las sanciones comerciales a sus competidores japoneses.

La corporación de los productores californianos afirma que con arreglo al acuerdo que en 1986 puso fin a la primera escaramuza de la batalla internacional del chip, la producción norteamericana debió haber obtenido durante 1987 una cuota equivalente al 12,3% del mercado total japonés como compensación de las ventas de chips japoneses en el mercado interno de USA. Según la SIA, la participación real de sus setenta y un socios en el mercado japonés no llegó a alcanzar el 10% durante el año pasado y la evolución en el actual muestra una tendencia declinante que se acerca a la más baja cifra histórica de participación: el 8,5% correspondiente al segundo semestre de 1986, que detonará el conflicto solucionado con el acuerdo que hoy se dice violado.

De acuerdo a la información de la SIA, las pérdidas de ventas que los productores de chips californianos soportan en razón del incumplimiento del acuerdo se hallan en el orden de los U$S 200.000 y la disminución de producción que ello apareja implica el desempleo de tres mil trabajadores.

Los productores americanos culpan del incumplimiento a las autoridades japonesas, a las que acusan de mantener un sistema tarifario discriminatorio para los chips de origen extranjero, que desalienta al empleo de semiconductores norteamericanos por parte de la industria electrónica e informática japonesa. Por ello, la SIA requiere de su propio gobierno medidas de realización que creen idénticas cargas al ingreso de los chips japoneses a los Estados Unidos y les permitan colocar en el mercado interno la producción que no es vendida en Japón por incumplimiento del acuerdo.

El petitorio de los productores de chips, elevado oficialmente por la SIA al gobierno norteamericano en marzo de este año, no ha originado hasta el momento otra reacción que prudentes comentarios por parte de los funcionarios gubernamentales del área de comercio y discretas pero firmes protestas oficiales japonesas. La presión actual del lobby del Silicon Valley difícilmente logre una definición de la cuestión, en plena época de definiciones electorales y cuando el presidente de la República aboga por un comercio exterior libre de regulaciones y trabas.

Entre tanto, el precio de los chips —que segun algunos observadores van camino de convertirse en otra de las "commodities" de cotización internacional— continúa creciendo y todo hace pensar que este episodio no será el último de la batalla por el dominio del mercado de tamaña dimensión e interés.

SADIO-ASINDOC Taller de Intercambio de experiencias en Informática y Documentación

En el marco de las XVII Jornadas Argentinas de Informática e Investigación Operativa (JAIIO), que se llevarán a cabo del 26 al 30 de Septiembre, la Asociación de Informática y Documentación SADIO—ASINDOC se encuentra abocada a la organización de un taller de intercabio de experiencias en Informática y Documentación, con los siguientes objetivos:

Propiciar el conocimiento directo y el intercambio de experiencias y desarrollos de informática y documentación. Promover a transferencia de tecnologías, la cooperación y la integración de esfuerzos y recursos. Delinear un diagnóstico de la informática y la documentación en el país. Posibilitar el contacto directo de especialistas de diferentes áreas temáticas y de diferentes regiones de país que ejercen su actividad en informática documentaria.

Temario:

Almacenamiento y recuperación de la información:

Creación de bases de datos documentales, factuales y referenciales. Aspectos de equipamientos y programación: en microcomputadoras y en equipos grandes. Formatos para el registro de datos. Desarrollo y uso de programas para aplicaciones especiales, por ejemplo, para gestión de tesauro, creación de directorios de becas, instituciones, proyectos de investigación, glosarios y otros documentos no convencionales.

Sistemas de apoyo para gestión de bibliotecas y Centros de documentación.

Circulación, préstamos, adquisiciones, registro de usarios, etc.

Compatibilidad de equipamientos, programas y formatos de registro de datos: Integración en redes, problemas de telecomunicaciones, etc.

Formación y capacitación de recursos humanos en Informática y Documentación

Informes e inscripción: Uruguay 252, (1015) Bs. As. Tel. 40-5755-45-3950



CONEXION EN LINEA

EL NEGOCIO TELEMATICO

Frecuentemente se considera que la telemática "abierta" paga, o sea aquella en que los usuarios pueden libremente adherirse sin restricciones, es un negocio vinculado a la informática. Este razonamiento se basa en una sobrevaloración del "medio" sobre el "mensaje" y es tan impropio como sería considerar que la edición de diarios es una parte del negocio de la fabricación de papel o de las máquinas de imprenta. Confusión más grave aún es creer que parte del "negocio" es la venta de equipos, básicamente de las terminales de usuario, y efectuar cálculos con esta hipótesis: la compra del equipo es un acto puntual e irrepetible, la rentabilización de la inversión sólo puede provenir de la utilización sistemática y del consiguiente pago del usuario. Finalmente y para terminar de complicar el tema, tiende a subvalorarse en la evaluación del proyecto el costo del contenido "intelectual", la información, los servicios, y de la comercialización de los mismos. En el negocio telemático hay tres actores principales: los proveedores de información, que en un sentido amplio no es sólo éste, sino también otros servicios, como las transacciones o la comunicación. Los proveedores de servicio, o sea las computadoras que alojan los servicios, los servidores, y por último los prestadores de comunicaciones, empresas telefónicas y redes de valor agregado. Solamente para éstas el "negocio" es general y no competitivo, el cobro del tránsito de la información por la red es independiente de quien la vende y la cobra y de la rentabilidad o no de ello. Por esto la acción de las compañias telefónicas es fundamental en el desarrollo del negocio. La provisión de servicios es por el contrario una variedad de los medios de comunicación, el medio telemático, y quienes están más próximos de él son las empresas de prensa escrita, radial o televisiva y las editoriales. Estas han sido en Europa y América del Norte, junto con los PTT, los impulsores del negocio telemático.

Parece en consecuencia que, para un desarrollo masivo de la telemática, hace falta por una parte una empresa de telecomunicaciones interesada en ampliar su facturación, por otra, medios y editoriales vigorosas y rentables, dispuestas a invertir en un nuevo negocio. Ambas condiciones no son por cierto las actuales de nuestro país, pero planes menos ambiciosos para una telemática "gran público" y "semi-profesional" deberían poder desarrollarse con un mínimo de apoyo o de no interferencia de los poderes públicos. Imaginación, pensamiento a mediano plazo y por supuesto una cierta dosis de riesgo son necesarios para los empresarios potenciales de este sector, pero la experiencia internacional demuestra que el negocio telemático no es una incógnita para el futuro sino una realidad cotidiana en los paises más avanzados.

Hasta la próxima

Roberto E. Escardó. Editor en Jefe.

BASES & DATOS

En las clasificaciones de las Bases de Datos suele excluirse sistemáticamente una categoria cada vez más importeante: las Bases de Datos de formas o estructuras. Estas tienen una generalizada aplicación en los sistemas de CAD-CAM, diseño y fabricación asistida por computadora, a través de la metodología conocida como "Tecnología de grupos". La misma se basa en la descripción morfológica y dimensional de piezas, de forma que pueda diseñarse una nueva a partir de la adaptación de otras existentes. La recuperación de la información necesita de un lenguaje de descripción a la vez sencillo y potente, sobre lo que ha conseguido importantes adelantos.

Las primeras aplicaciones en este campo en bases de datos de acceso público fueron los de recuperación de productos químicos orgánicos a través de su estructura, inaugurados a principios de esta década por QUESTEL en Francia, sobre la base Chemical Abstracts. Luego los productores de esta base, la American Chemical Society, lanzaron su propio sistema. Ambos se basan en un lenguaje de descripción simbólico-verbal. Las investigaciones se orientan hacia intefases de usuario que permitan la interrogación a partir de esquemas gráficos "libres". Un sistema en experimentación permite ya traducir una consulta de este tipo al lenguaje formal y utilizado actualmente. El mismo opera en una PC próxima al usuario y utiliza técnicas de inteligencia artificial. La falta de normas de manipulación y transmisión de gráficos y del manejo interactivo de éstos es una de las más serias restricciones al desarrollo de estas técnicas.

Un muy ambicioso proyecto, dada la complejidad de las estructuras involucradas, es la base de datos sobre aminoácidos que desarrollan varias universidades de Norteamérica. Parte importante del mismo es la investigación de los lenguajes de descripción, que permitan tanto la visualización de las estructuras como su recuperación.

Estrechamente ligados al campo del procesamiento de imágenes, estas técnicas deberán conocer un importante desarrollo en el futuro.

LOS ESTANDARES EN SERVICIOS TELEMATICOS

SEXTA PARTE

Luego de nuestro paseo lateral para conocer el Modelo ISO/OSI, y ya con los elementos para utilizarlo como referencia, vamos a tratar las normas del Nivel 6, el de presentación, postergando momentáneamente el tratamiento del Nivel 3, el de Red, y dejando de lado los Niveles 4 y 5, de Transporte y de Sesión respectivamente, donde hay pocos elementos de interés para el usuario de servicios telemáticos

En el Nivel 6, el de presentación, se encuentran por el contrario cuatro temas de gran importancia: Los Protocolos de Terminal, los de Transferencia de Archivos, los de Compresión de Datos y los de Seguridad de Datos: Encriptamientos y Autentificación. Cada uno de ellos, por su extensión e importancia, será tratado separadamente.

Los protocolos de terminal

Es frecuente, aún entre profesionales de la informática, referirse a una "Terminal ASCII" como si esta fuera una especie única y definida. Como ya vimos en la Segunda Parte de esta serie (MI Nº164) la misma denominación ASCII es por demás imprecisa. En realidad no existe una "Terminal ASCII", sino por lo menos una cincuentena de variedades principales, otro tanto de menos difundidas, incompatibles entre sí más allá de utilizar un alfabeto igual o similar y comunicaciones asincrónicas.

A las terminales asincrónicas, nombre genérico más apropiado, se las puede clasificar en tres grandes familias: las terminales modo rollo - la denominación en inglés, scroll, es la comunmente utilizada - las modo página y las modo formulario. En todos los casos, un factor de importancia es el tamaño de la pantalla, cantidad de filas o lineas y de posiciones por línea o columnas. Para las terminales de impresión, sólo el último factor, equivalente al ancho del carro, es relevante.

Terminales modo rollo

Se llama así a todas las terminales de impresión, en las cuales los caracteres recibidos son impresos en un rollo de papel; de allí el nombre genérico. Muchas con pantalla tienen este modo.

En las mismas, cuando llega un caracter, el mismo es desplegado en la pantalla o impreso, escribiéndose las columnas secuencialmente en cada línea. En el caso de terminales de pantalla, cuando la misma está completa y se pasa a la línea siguiente, las líneas existentes se desplazan una hacia arriba y la otra desaparece.

En el caso más sencillo, además de los caracteres representables, se utilizan unos pocos "caracteres de control", que al ser recibidos por la terminal causan efectos preestablecidos. Algunos están definidos por las Normas ASCII e ISO 6429. Usualmente se los denomina por sus símbolos, derivados de sus iniciales en inglés, por ejemplo: LF, Line Feed o Alimentación de Línea: Indica pasar a la línea siguiente; CR, Carriage Return o Retorno de Carro, indica al cursor o cabezal de impresión colocarse en la primera posición de la línea, VT, Vertical TAb o Tabulación Vertical, indica desplazarse a la misma columna de la siguiente fila. HT, Horizontal Tabulation, Tabulación Horizontal, indica desplazarse a la siguiente columna en la misma fila; BS, Back Space o Retroceso, indica retroceder a la comuna anterior en la misma fila.

Cuando se utilizan nada más que estos caracteres de control se tienen las terminales llamadas "Modo teletipo" o TTV, que son las más sencillas. Prácticamente todas las terminales asincrónicas son capaces de interpretar, aunque a veces con dificultades, este modo, mientras no se exceda la cantidad de columnas de una línea, por lo que muchas veces en la fase de conexión el servidor utiliza este modo, con un número reducido de columnas, generalmente veinte, hasta identificar a la terminal llamante.

Una importante limitación de las terminales en forma TTY es que en ellas no hay capacidad alguna de edición, es decir de modificar parte de la pantalla, ni de introducir datos en otro lado que en la última fila, ni de efectuar otra correción que eliminar el caracter a la izquierda o sobre el que se halla el cursor.

El modo TTY es frecuentemente utilizado por ser el más general, pero sus limitaciones en cuanto a la presentación de los datos, así como las de ingreso de información, conducen a presentaciones pobres, a lo que se suma una relativa ineficiencia en el envio de información.

Terminales modo página

Las terminales modo página tienen siempre pantalla, usualmente de 24 líneas y 40 u 80 columnas; tanto el operador como el servidor pueden mover el cursor a cualquier posición en la pantalla para ingresar o modificar información. La información usualmente es recibida en forma de pantalla completa. Una vez completada una y a pedido del operador o por comando del servidor, la misma es borrada y se envia la siguiente, pero en algunos casos se utiliza el scroll. El desplegado de la pantalla no es necesariamente secuencial, columna a columna dentro de cada fila, sino que puede hacerse en orden arbitrario.

En muchos casos las terminales modo página tienen capacidad de edición local, terminales en modo bloque, en las que una vez recibida una pantalla puede ser completada o modificada localmente; al estar la edición completa se transfiere la información al servidor en conjunto, generalmente oprimiendo una tecla ENTRE. En las terminales en modo página el conjunto de controles es más completo que en las de modo rollo, y en general los mismos no son caracteres aislados sino secuencias, o cadenas de caracteres, cuyas funciones son diversas, tales como, borrar la pantalla completa, o desde la posición del cursor hasta el fin de la pantalla o de la línea, colocar el cursor en una posición, fila y columna determinada, permitir insertar caracteres, etc.

Terminales modo formulario

Las terminales modo formulario se utilizan para aplicaciones específicas de consulta o transacción, tales como reserva de pasajes, verificación de tarjetas de crédito, punto de venta (POS) o los cajeros automáticos.

En su forma más simple, solamente puede introducirse una determinada cantidad de datos en campos fijos; la terminal tiene en general una inteligencia limitada, que le permite por ejemplo, validar que los valores ingresados en los campos sean del tipo correcto, numéricos o alfa-numéricos, o de longitud predetermina-

da. Los teclados a veces solamente comportan cifras y algunas teclas de función y las pantallas o displays son reducidos, a veces de una sola fila.

Terminales "bobas" versus terminales inteligentes

Frecuentemente se hace la distinción entre terminales "bobas" y terminales "inteligentes", la que es ciertamente confusa. En rigor, una terminal "boba" sólo es capaz de enviar al servidor los caracteres tipeados en el teclado y desplegar en la pantalla los recibidos del servidor. (Figura 1). Los caracteres enviados sólo son desplegados en la pantalla al recibirse en "eco" del servidor. Aún en este caso elemental es necesaria una cierta inteligencia para interpretar en el flujo de datos entrantes que caracteres son representables y cuales de control y de implementar las repectivas secuencias. Además, en general algunos parámetros son configurables, tales como la velocidad de comunicación como el modem o el alfabeto utilizado, lo que también requiere cierta inteligencia. En el otro extremo, una terminal "inteligente" es capaz de asumir ciertas funciones de comunicación en forma autónoma, por ejemplo, decidir que información enviar, o utilizar fomularios almacenados para la entrada de la información o permitir la interactuación entre una información recibida y otra almacenada o generar por si mismo y en forma autónoma conexiones para recabar o enviar información. Los límites entre ambas definiciones son por demás imprecisos y muchas veces se prestan a confusiones; por ejemplo una PC, de por sí "inteligente" cuando tiene un programa de comunicaciones trabajando en forma interactiva es una terminal totalmente "boba". Por el contrario , funciones tales como la transferencia de archivos, requieren inteligencia.

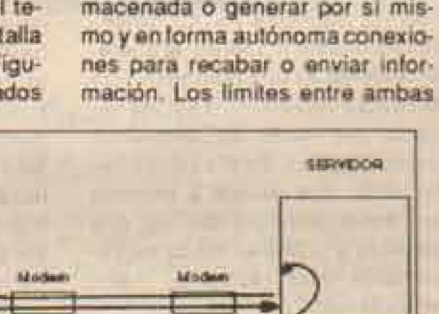


Figura 1

Juego de caracteres y atributos de visualización

En la segunda parte de esta serie y al referirnos a los alfabetos, habíamos tratado someramente el tema de los juego de caracteres o sea la cantidad de elementos distintos que pueden "representarse" en la pantalla. Unido a ello se halla el de los caracteres y secuencias de control que se utilizan para "ubicar" a los mismos en una posición determinada de la pantalla o para modificar la misma y para darles a los mismos "atributos" de visualización que modifican su presentación. Estos usualmente son, -en pantallas monocromáticas- el sobrebrillo, en la inversión en video o de fondo, el caracter toma el color del fondo y viceversa, el subrayado y el parpadeo. En muchos casos estos atributos pueden combinarse entre sí, por ejemplo: video invertido, subrayado y parpadeante. Algunas terminales admiten color, o de tonos de gris, a los atributos anteriores deben agregarse los colores de caracter y de fondo. El conjunto del juego de caracteres, los atributos y los caracteres de control forman lo que se denomina un protocolo de pantalla o representación.

En el próximo número continuaremos con este tema.

SECCION GUIA-SOFT

EMPRESAS

1 - **M Y A - MODELOS Y APLICACIONES EN COMPUTACION S.A.** - Av. Córdoba 1247, 2º P. "C" -1055- Capital- Tel:393-3128/0197.
2 - **SINDEC - SERVICIO INTEGRAL DE COMPUTACION S.R.L.** - Montevideo 536, 4º P. "H" -1019 - Capital - Tel: 46-6310
3 - **Ce De Se SISTEMAS** - Av. Rivadavia 2450, 4º "A" -1034- Capital- Tel: 47-1805/48-3954.
4 - **THERA S.A. SISTEMAS** - Varronte 1167, 7º P.- 1053- Capital- Tel: 45- 4191.
5 - **TARGET S.A.** - Salta1838- 1137- Capital- Tel: 23-8752/8721.
6 - **CONSAD S.A.** - Av. Córdoba 836 13º P.., of.1301- 1054- Capital- Tel: 393-3336/3308/3368.
7 - **AUTOM S.A.** - Sánchez de Bustamante 2516 PB "D"- 1425- Capital- Tel: 802-9913.
8 - **R & D S.A.** - Lavalle 1616, 3º Piso -1048 - Capital Federal - Tel: 46-6881/2 49-7250
9 - **COMPUSISTEM S.A.** - Tres Sargentos 463, 5º Piso 1054 - Capital Federal - Tel.: 313-2577/2584

INDICE TEMATICO	EMPRESA Nº	INDICE TEMATICO	EMPRESA Nº
AHORRO PREVIO	9	BASE DE DATOS	1/7/8
ASESORAMIENTO	6/9	BCRA	1/2
AUTOCRON	7	BIENES DE USO	1
AUTODIAL	7	BOLSA	2/4
AUTOFACT	7	BONOS	2/4
AUTOFILE	7	CAJA DE AHORRO	5
AUTOLINK	7	CAPACITACION	8
AUTOLOCK	7	CLIENTES	5
AUTOMAIL	7	COMERCIO EXTERIOR	3
AUTOPAGO	7	COMPRAS	7
AUTOSTAT	7	CONTABILIDAD	4/5
AUTOTEXT	7	CORREO ELECTRONICO	8

INDICE TEMATICO	EMPRESA Nº
CUENTAS CORRIENTES	5/7
CHEQUES	2
DBASE	7
DIVIDENDOS	4
EXPORTACION	3
FACTURACION	7
FONDO DE DESEMPLEO	5
GOBIERNO	6
HARDWARE	9
IBM	4
IBM SERIES/1	1
IBM 30XX	8
IBM 43XX	8/9
IBM 93XX	8
IBM S/3X	9
IMPORTACION	3
IMPRESORAS	9
IMPUESTOS	7
INFORMACION A BCRA	1
INFORMATICA ADMINISTRATIVA-CONTABLE	6
INFORMATICA BANCARIA	6
INFORMATICA DEL AGRO	6
INVENTERIOS	1
INVESTIGACION OPERATIVA	6
LENGUAJE DE 4º GENERACION	8
LOTUS 1-2-3	7
MAILING	7
MANOF SYSTEMS	1
MBS/2	1
MESA DE DINERO	2
MODELOS	6
MODIFILE	7
MONEDA EXTRANJERA	3
MULTIPLAN	7
MUNICIPALIDADES	6
PC	4/5/78
PC/BANK	5
PC/BANK (INTERAGENCIA)	5
PLAQUETAS	9
PLAZO FIJO	5
PRESTAMOS	5
QANTEL	5
RECAUDACIONES	5
RECURSOS HUMANOS	6
REDES	1/5
REDES LOCALES	8
REGISTROS DE DATOS	5
REVALUO	1/4
REVIN	1
SEGURIDAD	6
SICAP	9
SOFTWARE A MEDIDA	6
SOFTWARE DE COMERCIO EXT.	3
STOCK	7
SUCURSALES	5
SUELDOS Y JORNALES	7

INDICE TEMATICO	EMPRESA Nº
SWIFT	1
TELEPROCESAMIENTO	1/5/6
TELEX	1
TERMINALES	9
TERMINAL DE AUTOCONSULTA	5
TERMINALES FINANCIERAS	5
TERMINALES INTELIGENTES	5
TESORERIA	5
TITULOS	4
TITULOS PUBLICOS	2/4
TRANSF. ELECTR. DE FONDOS	5
VENTA DE HARDWARE	
VERYFILE	
VM/DOS	

R&D S.A.
Lavalle 1616, 3º P
1048-Capital Federal
Tel.: 46-6881/2 49-7250

DPS 8 - SOFTWARE: BASE DE DATOS RELACIONAL: DATACOM/DB:Sistema de Administración de Base de Datos Relacional de Alta Performance.ADR/DATA DICTIONARY: Diccionario Activo de Datos. ADR/DATA QUERY: Lenguaje de Consulta Para Usuarios Finales.VSAM/DL-1 y TOTAL TRANSPARENCIES: Software de Migración. ADR/IDEAL: Sistema de Desarrollo de Aplicaciones de 4ta. Generación. ADR/ROSCOE y ADR/VOLLIE: Sistema On-Line para desarrollo de programas. ADR/The LIBRARIAN: Adminstración de Programas Fuente y Biblioteca. ADR/LOOK: Monitor de Performance Integral. ADR/eMAIL: Correo Electrónico. Inforeach: Base de Datos Distribuidas. MINDOVER: Sistema Experto de Performance y Tuning para MVS.
Extensiones de PC de todos los productos. Estos productos están disponibles para los equipos IBM 43xx, 30xx, 93xx y compatibles. Bajo los sistemas operativos: VM. DOS/VS al VSE/SP y OS/VS1 al MVS/XA.

COMPUSISTEM S.A.
Tres Sargentos 463 5º P
1054 Capital Federal
Tel. 313-2577/2584
DPS 9-DIVISION SERVICIOS CONSULTORIAPARACIRCULOS DE AHORRO PREVIO

Area Institucional: Constitución de Sociedades Administradoras. Preparación de Bases Técnicas.
Areas Operacionales: Definición de estructura. Redacción de Manual de Procedimientos. Diseño de formularios.
Area contable: Confección del Plan de Cuentas. Redacción de Manual de Imputaciones.
Area Comercial: Definición de la estrategia de Comercialización. Plan de capacitación.
Area informática: Organización o reestructuración del Departamento de Computación. Relevamiento, diagnóstico y recomendaciones para optimizar el Departamento de Computación. Definición de estructura, recursos y procedimientos. Auditoría de sistemas.

COMPUTACION PARA CIRCULOS DE AHORRO PREVIO

Software: Licencia del uso del paquete de programas SICAP. Desarrollo de programas o nuevos módulos de los sistemas actualmente instalados en computadoras IBM S/36 o 4300.
Procesamiento: Servicio de procesamiento de datos con el sistema SICAP cuyas principales características son: módulos específicos para toda la operatoria de Ahorro Previo, desde la suscripción hasta la liquidación de los grupos, en sus variantes de Bienes, Dinero e Inmuebles. Información Gerencial Instantánea para toma de decisiones comerciales y financieras. Contabilidad interactiva con input automático de transacciones generadas por el sistema. Acto de adjudicación y data entry interactivo con terminales on-line o soportes de PC stand-alone

AUTOM S.R.L.

S. de Bustamante 2516, P.B. "D"
1425 Capital Federal
Tel. 892-9913

DPS 7 - Primera empresa argentina dedicada con exclusividad absoluta al desarrollo, edición y venta de software "standard" para Computadoras Personales. No se hacen desarrollos " a medida", ni adaptaciones para usuarios individuales. Esta política exige productos de gran flexibilidad, totalmente parametrizables.
Son conocidos los casos de AUTOFACT (Facturación-Stock-CC) ym AUTOPAGO (Sueldos), cuya total libertad de configuración (archivos, informes, moneda,conceptos, etc.). Permite utilizar la versión "standard" aregntina en otros países de habla hispana.
Todos los productos vendidos por AUTOM presentan una característica en común: utilizan archivos formato AUTOFILE. Este simple detalle permite el fácil intercambio de datos entre todos los módulos, formando un sistema integrado, de gran versatilidad.

Lo que sigue es una breve descripción de cada módulo:

1)AUTOFILE
Administrador de archivos. Generador de Informes con Procesador de Lenguaje Natural (NLP)
2) AUTOFACT
Facturación/Stock/CC. Incluye compras, pagos a proveedores, CC deudores y acreedoras y cartera de documentos.
3) AUTOPAGO
Sueldo y Jornales. Libre definición de conceptos, fórmulas, moneda, impuestos, acumuladores, listados.
4) AUTOSTAT
Expresa los datos de cualquier archivo AUTOFILE en forma de gráfico de barras, histograma, acumulados, tendencias.
5) AUTOMAIL
Imprime etiquetas (formato y condiciones definibles) con datos de cualquier archivo AUTOFILE.
6) AUTOTEXT
Editor de textos.Puede insertar (merge) datos de archivo AUTOFILE. Salida para impresora o modem.
7) MODYFILE
Comunicación interarchivos. Permite la migración de datos de un archivo a otro, modificando su estructura.
8) AUTOLINK
Comunicación a Multiplan/Lotus 1-2-3/Dbase. Transforma AUTOFILE a formato DIF, y viceversa.
9) AUTODIAL
Busca números telefónicos en archivos AUTOFILE, y realiza el discado por el port RS-232. No requiere modem.
10) VERYFILE
Analizador inteligente de archivos. Detecta errores de tipeo o datos numéricos mal ingresados.
11) AUTOCRON
Agenda/Calculadora/Hora internacional/Calculadora. Datos de la agenda formato AUTOFILE.
12) AUTOLOCK
Protección de datos confidenciales en Winchester. Accesos con "password" para diferentes usuarios.

LA INFORMATICA "VERDE"

Robots que ordeñan vacas, computadoras que deciden cuando y en qué medida alimentar los animales, sensores automáticos que indican la temperatura de los bovinos y en caso de enfermedad aislan la leche infectada, pronosticadores meteorológicos que ofrecen a los agricultores la situación climática de por lo menos una semana y computadoras que administran planes de siembra e irrigación de los campos.

La agricultura electrónica ofrece constantemente nuevos servicios. Tanto es así que algunos pronosticaron la inminente llegada de una nueva era para el sector. Los programas de computación están de hecho sustituyendo poco a poco todas las labores manuales y transformando a los agricultores en expertos en informática.

En Gran Bretaña existen ya doscientos cuarenta establecimientos que trabajan con el auxilio de sistemas informáticos. En ellos, la alimentación y el pesaje automático de las cabezas de ganado se realizan directamente en recintos interiores y a través de la computadora, el criador puede controlar, incluso, los precios de venta actualizados de los animales.

En Italia, aunque más lentamente, se trata de poner la informática al servicio de la agricultura. Hace un par de meses atrás, por ejemplo, que se anunció el nacimiento de Gea (Gestión Agrícola), una compañía que se constituyó en Ferrara con intervención de la administración provincial, de la Cámara de Comercio y de las asociaciones del sector, con el propósito de promover la divulgacion de informaciones y conocimientos en el campo agrícola.

Gea estará pronto vinculada a algunos bancos de datos europeos (como el inglés Ag-News y la Messagerie Electronique de la CEE) y suministrará noticias sobre precios y mercados de diversos países europeos. También próximamente Gea estará en condiciones de efectuar reconocimientos sistemáticos vía satélite del estado de la agricultura en la región de Ferrara.

También en el campo del telerelevamiento, Italia está dando los primeros pasos . En los Estados Unidos existen ya enormes fincas agrícolas donde la "teleagricultura" se practica en gran estilo. En Oregón, por ejemplo, la Eastern Oregon Farming Company emplea desde hace casi tres años sistemas de telerelevamiento para cultivar sus campos propios: innumerables sembrados de papas, cereales y alfalfa que redituan trece millones de dólares anuales.

En Italia, donde los establecimientos agrícolas son de dimensiones mucho menores, el telerelevamiento es una operación que se realiza sobre todo a escala nacional por medio de compañías especializadas. Es el caso de Telespazio, que recoge y procesa los datos provenientes de dos satélites puestos en órbita expresamente para recolectar datos sobre el estado de los suelos.

Al servicio de los pequeños agricultores se encuentran ciertas organizaciones regionales como Ersal (Ente regional de desarrollo de Lombardía), que está efectuando el censo de todos los puntos de relevancia metereológica de la región y en breve habilitará nuevas estaciones Agrometereológicas automáticas y un centro de procesamiento de datos para la producción de informaciones para el usuario agricultor.

También los tambos italianos se informatizan poco a poco. Los más avanzados se encuentran en Emilia y Lombardía. Hoy no solamente es posible ordeñar hasta 150 animales con un aparato (mientras tan solo ayer un hombre conseguía atender únicamente quince), sino que se puede mantener bajo riguroso control a todos los animales. En algunos tambos, por ejemplo, cada vaca lleva un collar electrónico. Tan pronto se acerca al comedero se la "reconoce" y se le da la ración que le toca: abundante si da mucha leche, escasa si hace poca. Y a la que no come la ración adecuada, se la manda a pastoreo.

Impreso en sistema Offset por Talleres Gráficos Alemann S.R.L. 25 de Mayo 626 Bs. As.

Tarifa Reducida
Concesión Nº 3849